COLLABORATIVE ROBOTS • ROBOT DINOSAURS • BIONICS

SERVO MAGAZINE

FOR THE ROBOT EXPERIMENTER

www.servomagazine.com
DECEMBER 2006

MEET DARwIn

The Evolution of an Intelligent Anthropomorphic Robot

- Build a Twin Motor Driver
- The Great Serial Port Caper
- PROTOBot: Amoeba
- Beginner's Robotics On $50 A Month
- Interface Your PC To An R/C Radio

JAMECO® ROBOT STORE

Perfect holiday gifts for kids of all ages!

robotic kits

chassis

servos

passives

optos

integrated circuits

semiconductors

muscle wires

connectors

motors

test equipment

software

books

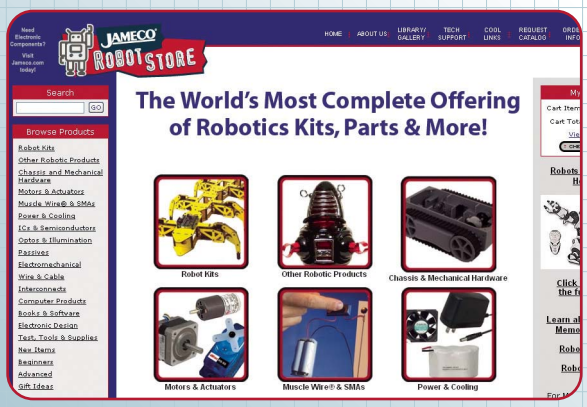
Robotic Kits and Components...
...there's something for everyone!

Robotic kits help you and your child to experience and learn about perception and control using a variety of sensors and actuators. Challenge yourself with Jameco's selection of fun and interactive kits! You or your child can assemble the kits and then enjoy endless hours of discovery.

Check out our unique selection of robotic kits at **www.RobotStore.com!**

- **Robot Insects & Animals**
- **Solar Robots**
- **Listening, Touching & Seeing Robots**
- **Hackable Robots**
- **Programmable Robots**
- **Educational Kits**
- **Legged and Wheeled Platforms**
- **OctoBot Survivor Kit**

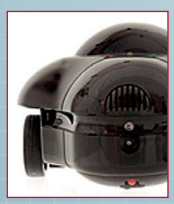
At Jameco's RobotStore you can get the world's most complete robotic offering—*all in one place!* You'll find kits for all ages and experience levels, along with gear boxes, servos and chassis, for those who are ready to take on the extreme.

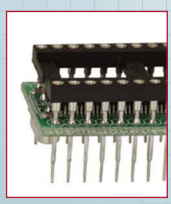


WWW.ROBOTSTORE.COM/SRC

1-800-374-5764

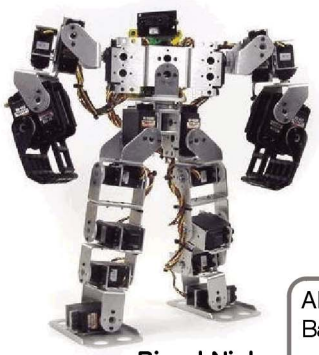
Call for your free catalog today!





Lynxmotion

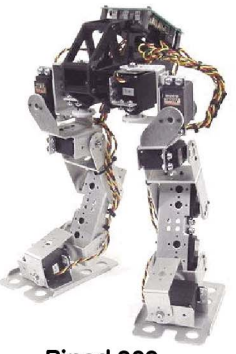
The Lynxmotion Servo Erector Set
Imagine it... Build it... Control it!



Biped Nick

Biped Pete

Biped Scout



Biped 209

Featured Robot

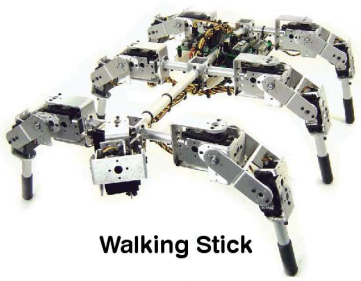
Aluminum Legs!
Ball Bearings!

RC or Autonomous!
On-line Video!

Height: 7"
Width: 21"
Weight: 56oz.

CH3-R Hexapod!

The CH3-R is the coolest looking 3DOF hexapod! Watch it move and you will realize it's also the most advanced! It can walk in any direction (translate), turn in place (rotate), and any combination of the two at variable rates, plus much more.



Walking Stick

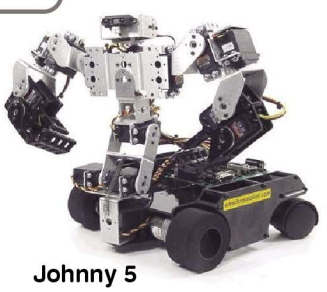
With our popular Servo Erector Set you can easily build and control the robot of your dreams!
Our interchangeable aluminum brackets, hubs, and tubing make the ultimate in precision mechanical assemblies possible. The images here are just a sample of what can be built. The Bot Board and SSC-32 provide powerful control options. Our Visual Sequencer program provides powerful PC, Basic Atom, or BS2 based control.

Bot Board - $24.95
Carrier for Atom / Pro, BS2, etc.
Servo and Logic power inputs.
5vdc 250mA LDO Regulator.
Buffered Speaker.
Sony PS2 game controller port.
3 Push button / LED interface.

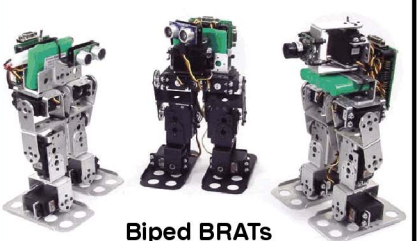
SSC-32 - $39.95
32 Channel Servo Controller.
Speed, Timed, or Group moves.
Servo and Logic power inputs.
5vdc 250mA LDO Regulator.
TTL or RS-232 Serial Comms.
No better SSC value anywhere!

We also carry motors, wheels, hubs, batteries, chargers, servos, sensors, RC radios, pillow blocks, hardware, etc!

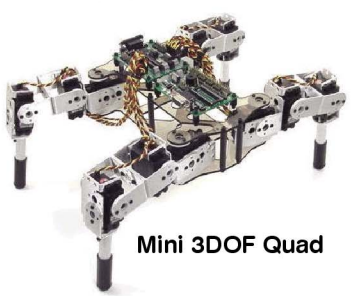
Visit our huge website to see our complete line of Aluminum and Lexan based robot kits, electronics, and mechanical components.



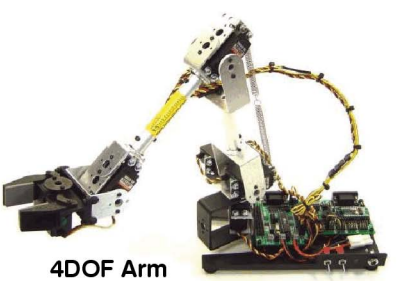
Johnny 5



Biped BRATs



Mini 3DOF Quad



4DOF Arm

Images represent a fraction of what can be made! **www.lynxmotion.com** SES now has 52 unique aluminum components!

SERVO MAGAZINE

Columns

08 **Robytes** *by Jeff Eckert*
Stimulating Robot Tidbits

10 **GeerHead** *by David Geer*
Robot Dinosaurs Come Alive and Thrive!

14 **Lessons From the Lab**
by James Isom with Brian Davis
NXT Robotics: Remote Control

19 **Ask Mr. Roboto** *by Pete Miles*
Your Problems Solved Here

72 **Robotics Resources**
by Gordon McComb
Sound Generation Techniques for Robots

75 **Rubberbands and Baling Wire** *by Jack Buffington*
The Great Serial Port Caper

78 **Robotic Trends** *by Dan Kara*
Educational Robotics is the Smart Choice

84 **Appetizer** *by Robin Hewitt*
An Invitation to Computer Vision with OpenCV

87 **Then and Now** *by Tom Carroll*
Bionics — Where Robots Meet Human Flesh

ENTER WITH CAUTION!
26 The Combat Zone

Page 08

Page 10

Page 87

***SERVO** Magazine* **(ISSN 1546-0592/CDN Pub Agree#40702530)** is published monthly for $24.95 per year by T & L Publications, Inc., 430 Princeland Court, Corona, CA 92879. PERIODICALS POSTAGE PAID AT CORONA, CA AND AT ADDITIONAL ENTRY MAILING OFFICES. POSTMASTER: Send address changes to **SERVO Magazine, P.O. Box 15277, North Hollywood, CA 91615** or Station A, P.O. Box 54, Windsor ON N9A 6J5; cpcreturns@servomagazine.com

12.2006
VOL. 4 NO. 12

Features & Projects

33 **Collaborative Robotics**
by Bryan Bergeron
An introduction to collaborative robotics and the key implementation challenges readers should consider.

40 **DARwIn**
by Karl Muecke, Patrick Cox, and Dennis Hong
Part 1: Concept and General Overview of a Dynamic Anthropomorphic Robot with Intelligence.

44 **Interface Your PC to an R/C Radio**
by Michael Simpson
Create a very accurate and reliable interface to any R/C radio with up to six channels.

50 **PROTOBot: Amoeba!**
by Camp Peavy
A complete interactive robot.

57 **ROBOGames Prep**
by Dave Calkins
This month: Firefighting robots.

60 **Building a Twin Motor Driver**
by Peter Best
Discover a cool twin-motor hardware design and learn how to code motion drivers.

67 **Beginner's Robotics on $50 a Month**
by Paul Pawelski
Getting started in robotics doesn't have to cost you an arm and a leg.

Departments

06 Mind/Iron

07 Bio-Feedback

18 Events Calendar

22 New Products

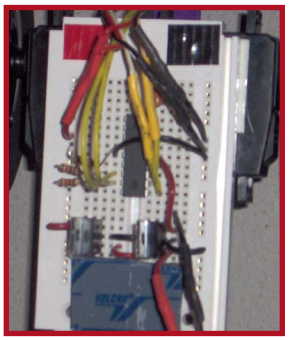
24 Menagerie

48 Robotics Showcase

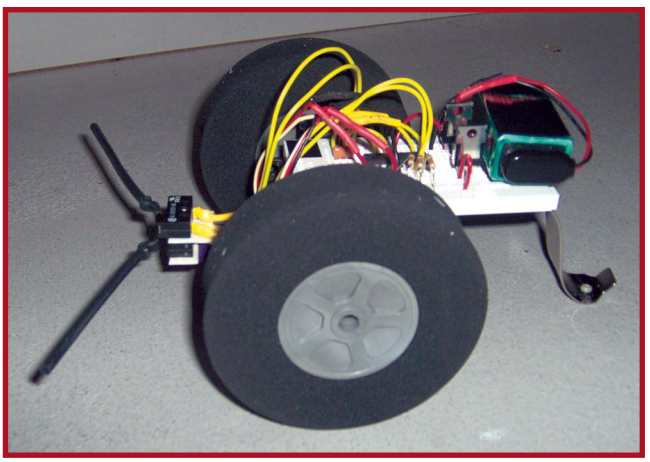
81 Robo-Links

82 *SERVO* Bookstore

90 Advertiser's Index



Page 50



Mind / Iron

by Camp Peavy

Failure is the path of least persistence. — George M. Van Valkenburg, Jr. (1938-)

Robot building is hard work. It is an interdisciplinary craft requiring expertise in mechanics, electronics, and programming; each field deep and wide in and of itself; each has that "the-more-you-know-the-less-you know" quality ... where every answer creates two new questions and what's the point, after all?

Sometimes things turn out better than expected and inspiration builds upon inspiration. Other times (mostly), things don't work out as planned; both are reasons to have tried, though (yes, Yoda, there is "try"). Otherwise, you wouldn't know.

When something doesn't work or you don't have the right part, answer, or financing, work on another aspect of the project. At least you know what does *not* work. Enter a contest! This is a real motivator. Nothing like a deadline to force you to create. When I was building "Autonomous Rodney" for the 1996 Robot Wars, I couldn't get the optical-based passive wheel disc encoder working and was running out of time (real robot builders work best under pressure). Then it happened in the security section of a RadioShack ... Epiphany — magnets and a reed switch! Worked like a charm ... even in the dust at Burning Man. The point is I had dreamed of building this type of robot for years but probably would have never finished it if I didn't enter that contest. The best laboratory is the real world and many times we need a deadline to force us to create. The more you get your robot out and demonstrating it, the better it gets. Build, test, and demonstrate ... too many folks are trying to learn everything in the world before actually doing anything physical. Build, test, and demonstrate. You learn through your fingers.

What's the next step in your robotic project? Are you in the planning stage? Are you "finished?" If so, show it to someone. Be prepared that some might not be impressed with your new gizmo, but they are missing the point of the entire journey. Others will get a kick out of it no matter how inane. Start another project ... or add to what you have. Robot building is an iterative process. You build upon what you have built. Another problem ... "The spirit is willing but the flesh is weak." Everyone dreams of building a robot but actually doing it is too much work. Let me say this ... robot building is one of the most important things you can do in life and has more potential payoff than anything. Enjoy the pastime and find others with the same interest.

We have had industrial robots for decades now. The new thing is mobility. We will soon be entering an age of smart machines where devices will know where they are and will eventually pick and place in a cluttered environment with great dexterity. "No way" you say? Let's pretend we're in the year 1900 and I was telling you about the 1950s ... cars, airplanes, telephones, etc. "Not possible!" you would say. Now let's suppose we're in the 1950s and I'm telling you about technology in the year 2000 with the Internet, cell phones, microwave ovens, Global Positioning Systems, etc.

Mind/Iron Continued →

Published Monthly By
T & L Publications, Inc.
430 Princeland Court
Corona, CA 92879-1300
(951) 371-8497
FAX **(951) 371-3052**
Product Order Line **1-800-783-4624**
www.servomagazine.com

Subscriptions
Inside US **1-877-525-2539**
Outside US **1-818-487-4545**
P.O. Box 15277
North Hollywood, CA 91615

PUBLISHER
Larry Lemieux
publisher@servomagazine.com

ASSOCIATE PUBLISHER/ VP OF SALES/MARKETING
Robin Lemieux
display@servomagazine.com

CONTRIBUTING EDITORS
Jeff Eckert, Tom Carroll, Gordon McComb, David Geer, Pete Miles, Kevin Berry, Dave Calkins, Bryan Bergeron, Karl Muecke, Patrick Cox, Dennis Hong, Brian Benson, Tim Wolter, Leonard Ginn, Camp Peavy, Randy Hootman, Jack Buffington, Dan Kara, Robin Hewitt, James Isom, Brian Davis, Peter Best, Michael Simpson, Paul Pawelski, R. Steven Rainwater

CIRCULATION DIRECTOR
Tracy Kerley
subscribe@servomagazine.com

WEB CONTENT/STORE
Michael Kaudze
sales@servomagazine.com

PRODUCTION/GRAPHICS
Shannon Lemieux

ADMINISTRATIVE ASSISTANT
Debbie Stauffacher

Copyright 2006 by
T & L Publications, Inc.
All Rights Reserved

All advertising is subject to publisher's approval. We are not responsible for mistakes, misprints, or typographical errors. *SERVO Magazine* assumes no responsibility for the availability or condition of advertised items or for the honesty of the advertiser. The publisher makes no claims for the legality of any item advertised in *SERVO*. This is the sole responsibility of the advertiser. Advertisers and their agencies agree to indemnify and protect the publisher from any and all claims, action, or expense arising from advertising placed in *SERVO*. Please send all editorial correspondence, UPS, overnight mail, and artwork to: **430 Princeland Court, Corona, CA 92879.**

BIO-FEEDBACK

Katherine Claire Miles was born October 4th to proud parents Pete and Kristina. Weighing in at a healthy 9 pounds, 1 ounce, she was a meager 20-1/4 inches long.
You done good, Mr. & Mrs. Roboto!

Did you know that if you're a paid subscriber to *SERVO Magazine*, you can get the online version for *FREE*?
Go to www.servomagazine.com

... once again, the inventions seem like science-fiction.

Is the pace of technical development slowing? Do you think there will still be technical obstacles to fully developed humanoid robots in 50 years? Do you think people won't need fully developed humanoid robots? Consider the aging populations of the developed countries. Who's going to do the grunt work of the future? Eventually, the robotics industry will be larger than the computer industry. If you go to the "Computer History Museum" in Mountain View, CA (Silicon Valley) and follow the evolution of historical computing equipment, it ends up with robots. There will eventually be a robotics age on par with today's computer age.

The end game for robotics is nothing less than a humanoid slave. Indeed, it is the origin of the word, as "robot"comes from the Czech "robota" or forced labor. Robots are our progeny. They are the next stage in evolution. By 2050, we will have C3PO type androids ... it is inevitable. If you aren't building robots today, you are missing out on all the fun and the other rewards that will inevitably follow.

Mwa-ha-haaa! SV

Dear SERVO:

Regarding a recent Mr. Roboto topic ... I would like to correct an apparently widely-held misconception about the transmitted signal of an RC system. The desired servo position is NOT determined by the duration/width of a transmitted pulse. The desired servo position is determined by the position of a constant width pulse. That is why it is called Pulse Position Modulation. The transmitted pulse train consists of a start pulse and one pulse for each channel. A three-channel system has four pulses in a frame of data. Each pulse is of fixed width, typically 0.25 ms. In AM systems, the pulse turns the carrier wave off. Thus, the carrier is on for the majority of the time, which helps keep the receiver's AGC (automatic gain control) happy. In FM systems, the pulse may increase or decrease the carrier frequency depending on the brand of the radio.

The position of the pulse — which determines the position of the servo output — is measured with respect to the previous pulse. The position of the current pulse is the distance/time between the leading (or trailing) edge of the current pulse and the leading (or trailing) edge of the previous pulse. The use of pulse position rather than pulse width minimizes the effect of pulse distortion and long rise and fall times. The transmitted pulses are not text-book square pulses. The rise and fall times are intentionally increased to meet FCC mandated bandwidth limits. With long rise and fall times, the pulse is wider near the base than it is near its top. Thus, the width is somewhat ambiguous and the measured width may depend on the signal strength. Since the position of a pulse is the distance between a point on the current pulse and the corresponding point on the previous pulse, the shape of the pulse has little effect on the measured position.

The decoder in the receiver separates and converts this PPM pulse train into individual Pulse Width Modulated (PWM) pulse trains for each servo. In some receivers, the decoder is a serial-in parallel-out shift register. Because the transmitted signal is PPM, there is no delay between channels. The pulse for channel 2 starts at the same time that the channel 1 pulse ends, no gap. This can be verified by displaying adjacent servo channels on a dual channel oscilloscope.

Will Kuhnle
Lavon, TX

Writer response:

Thanks for the information. This is a bit different from what I have been taught. Thanks for pointing these specifics out. Perhaps you would be willing to put an article together for SERVO to illustrate these specifics so that this widely-held misconception can be corrected. I personally would love to see it. — Pete Miles

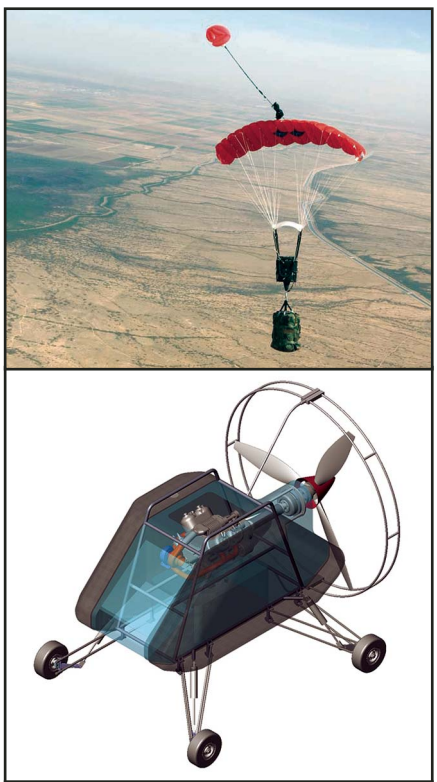
Robytes

by Jeff Eckert

Are you an avid Internet surfer who came across something cool that we all need to see? Are you on an interesting R&D group and want to share what you're developing? Then send me an email! To submit related press releases and news items, please visit www.jkeckert.com

— Jeff Eckert

New UAVs Demonstrated



The Atair Onyx (top) and LEAPP (bottom) systems. Photos courtesy of Atair Aerospace.

UAVs for military operations are becoming ever more common, and Atair Aerospace (**www.atairaerospace.com**) demonstrated two new ones at the recent Association of the US Army (AUSA) annual meeting and exposition. First up was the Onyx™ precision guided parachute system — a parafoil designed to carry cargo from altitudes up to 35,000 feet, glide autonomously for better than 30 mi, and land on a target with accuracy of about 150 feet. It combines adaptive control, flocking/swarming, and active collision avoidance capabilities to allow multiple systems (50 or more) to work simultaneously in the same airspace and deliver up to 2,200 lbs of "mission critical supplies."

The nature of such supplies can vary, but a hint is that the Onyx is routinely referred to as a "smart bomb." Also demonstrated was a scaled-back version of Atair's Long Endurance Autonomous Powered Paraglider (LEAPP). The Micro LEAPP, which can function autonomously or via remote control, is designed for special operations intelligence, surveillance, and reconnaissance (ISR) missions that involve up to eight hours of flight time and a maximum payload of 50 lbs. (Its big brother can spend up to 55 hours aloft, carry up to 2,400 lbs beneath its 112-foot wingspan.)

Automatic Refueling Developed

Also having obvious military implications is the Autonomous Airborne Refueling Demonstration (AARD) system, developed by the Defense Advanced Research Projects Agency (DARPA, **www.darpa.mil**) and NASA's Dryden Flight Research Center (**www.nasa.gov/centers/dryden/**). Built on GPS-based relative navigation and an optical tracker, it provides the precise positioning needed to drop a refueling probe into a 32-in basket that dangles in the airstream behind a tanker (in this case, a Boeing 707-300 operated by Omega Air Refueling Services) and drains into the F/A-18.

The AARD system allows autonomous refueling of airborne platforms. Photo courtesy of NASA, by Jim Ross.

Although the system initially made the connection in only two of six attempts, it safely recovered from each flub and completed its mission. In this demonstration, pilots were on board the F/A-18 for safety purposes, but the operation was carried out without their intervention.

Robotic Weed Killer

Thanks to the University of Illinois, this solar-powered bot will soon be controlling weeds in some experimental fields. Photo courtesy of U of I.

With the goal of reducing herbicide use, Lei Tian, an agricultural engineer at the University of Illinois (**www.uiuc.edu**), has developed a solar-powered robot that can track down weeds and then — using a robotic arm — cut and poison them on a close-up and personal basis. The machine — which moves at about 3 mph — uses GPS for navigation, plus it sports two small cameras to give it distance perception. An on-board Windows® computer allows it to decide what is a weed and what is not, and it has a wireless Internet connection for communications and an 80-GB drive for data storage. At present,

the robot is used only to combat weed infestation, but in the future, it may be fitted with different sensors and cameras that would allow it to examine soil properties or plant conditions. For now, the device will be used on an experimental basis, moving along crop rows in fields at the U of I, but commercial development seems feasible.

Motion Card Offers Reduced Cost

The Prodigy PC104 provides multiaxis motion control for a range of motors. Photo courtesy of Performance Motion Devices.

Moving back to the component level, Performance Motion Devices (**www.pmdcorp.com**) has introduced the Prodigy PC104 Motion Card for multiaxis, multimotor control. Available in one- through four-axis versions, its features include trajectory generation, servo loop closure, quadrature signal input, motor output signal generation, performance trace, on-the-fly changes, commutation, and other functions for DC brush and brushless DC, step, and microstepping.

The cards are programmed in C/C++ or Visual Basic and features include S-curve, trapezoidal, velocity contouring, electronic gearing, and user-generated profile modes. They accept input parameters such as position, velocity, acceleration, and jerk from the host and generate a corresponding trajectory. Instantaneous on-the-fly changes can be sent by the user, and external signal inputs can be used to program automatic profile changes. Communication is via a PC/104 bus, CANBUS, or serial port. Prices start at $380 in production quantities.

Care to Clone Yourself?

In a twist that becomes ever more twisted as you think about it, Hanson Robotics (**www.hansonrobotics.com**) and Direct Dimensions, Inc. (**www.directdimensions.com**), are developing a process through which they can perform a 3-D laser scan of someone's head, create a life-like reproduction of it, and plunk it down on top of a humanoid robot. Hanson has demonstrated robots that show a range of human expressions, including joy, sorrow, and surprise, so you can expect a fairly impressive level of realism. It is suggested that with this technology, "You can build yourself to comfort sick loved ones when you are unable to physically be there or design yourself for posterity." Sure, and you could also bring back your beloved Aunt Hilda to bake cookies for you. However, it doesn't take much imagination to picture Heidi Klum fetching a frosty martini, Barry Manilow cleaning your bathroom, or Rasputin singing "Happy Birthday" to you. Add a little more imagination, and we probably don't want to go there. SV

FaceScan technology will allow robots to be fitted with heads of yourself, loved ones, or famous people. Photo courtesy of Hanson Robotics.

HOBBY ENGINEERING
The technology builder's source for kits, components, supplies, tools, books and education.
Robot Kits For All Skill Levels
ICs, Transistors, Project Kits
Motors, Frame Components and Scratch Builder Supplies.
Books and Educational Kits
BEAM Kits and Components
Order by Internet, phone, fax or mail.
www.HobbyEngineering.com
1-866-ROBOT-50
1-866-762-6850
1-650-552-9925
1-650-259-9590 (fax)
sales@HobbyEngineering.com
180 El Camino Real
Millbrae, CA 94030
Visit our store near SFO!
Most orders ship the day received! World-wide shipping. Convenient payment options.

GEER HEAD

by David Geer

Contact the author at geercom@alltel.net

Robot Dinosaurs Come Alive and Thrive!

That's right, it's robot dinosaurs plural as UGOBE's Pleo and WowWee Robotics' Roboreptile make ready for arrival.

Pleo

Pleo — UGOBE's flagship robot — stands 6.82 inches tall with a width of 5.84 inches and a length of 18.8 inches. Playful, Pleo is capable of responding to its environment with hundreds of different emotional cues ranging anywhere from sadness to grumpiness to playfulness and everywhere in between.

UGOBE reserves the right to change any of the technical details of Pleo at any time.

Pleo is the sum of a wide assortment of technology including a 32-bit Atmel ARM 7 microprocessor, which is the delightful dino's primary brain. The prehistoric critter uses a 16-bit sub-processor, which is dedicated to a camera system's processing, which accomplishes the bot's eyes' image processing and bus translation.

Pleo also packs four eight-bit processors that enable low-level motor control for the robot's servos, as well as feedback for "derived" sensors. Speaking of sensors, Pleo is mighty sensitive, orchestrating 34 sensors in all including the camera.

Image of Pleo's remote control dash board.

FROM UGLY DINOSAUR-LING TO BEAUTIFUL ROBOT PET

During its early stages of development, Pleo wasn't always the handsome little dino-bot you see before you today. When UGOBE was developing its motion systems, Pleo was in a state of ... well, you could say he was having a bad hair month, but he didn't even have a head to have hair on.

In that state, Pleo consisted of partial body shells and framing with no skin, some over-used foam and glue, and a piece of duct tape at the end of its neck where its head should have been (the head was being repaired). As Pleo's creators put it, "Pleo, in this state, was about the ugliest robot you've ever seen. But when it started to play motions and show curiosity and emotion through its movements, it transformed from an ugly duckling into a pet — even as strange looking as it was. At this moment, it really hit UGOBE how powerful motion is to humans."

Specifically, the total sensor count includes:

• Four foot switches that detect footsteps up and down.

• Seven capacitive touch sensors for the four legs, back, shoulder, and head.

• A single derived white light sensor.

• Two microphones.

• 14-force sensors (one per servo).

• An orientation tilt sensor.

• An IR transceiver for bi-directional data communications.

• Another for detecting objects in Pleo's mouth (guess they knew kids would be sticking something in there!).

Additional sensors include those used to measure battery temperature and voltage. Most sensors are original designs from UGOBE in order to meet the specifications for size and compactness. Sensors and hardware empower Pleo's intelligence and behaviors. Through the many sensors listed here, Pleo can:

• Recognize objects for avoidance or interaction.

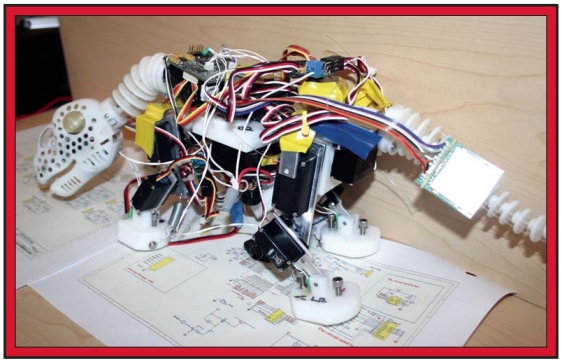
• Detect motion, light, and changes in light level.

• Detect the edges of objects in its environment.

• Detect sound and its source and direction.

• Recognize interactions — like being touched on its head, shoulders, back, legs, or feet.

• Recognize body position and spatial orientation and abuse via force feedback joints (be nice to your new Pleo, please; he may react if he doesn't like how he's being treated!).



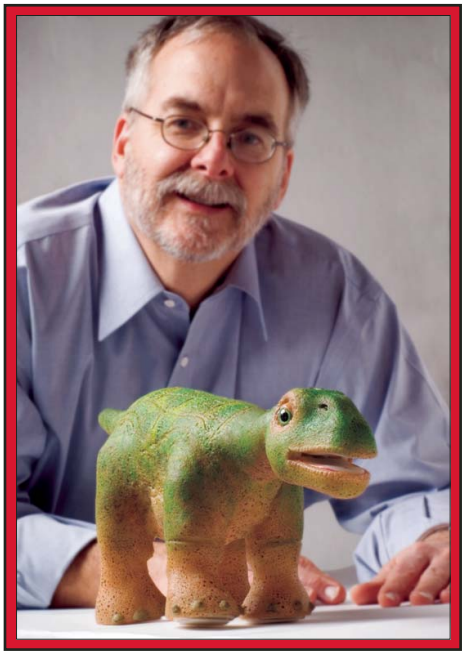
Pleo skinned alive with no skin and no toes.

• Recognition of and communication with other Pleos (which, of course, you can only experience if they sell you a lot of Pleos).

Pleo also uses 14 motors (standard, low voltage, DC), 150 gears and clutches, a rechargeable NiMH power pack, a USB port and connector, an SD/MMC slot, and software and systems programmed in a mix of C/C++, Assembly, and open source scripting languages (yet to be announced). Pleo employs USB communications and a standard file system for the SD/MMC card.

First Ever Robotics and Engineering Appearing in Pleo

In Pleo, UGOBE has combined life-like motion with a wide range of flexibility of movement, which correspond with and respond in relation to Pleo's emotional states and well-being of the moment. This is unique in the robotic space.



John Sosoka of UGOBE with Pleo prototype.

What, When, Where, and How Much?

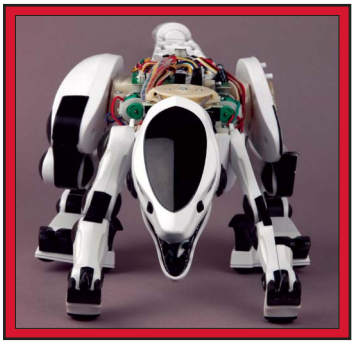
UGOBE's Pleo — a new entry in the robot reptile market — will be available through online pre-orders starting in the midst of the holiday shopping season — depending on your particular holiday — on December 24th. In March, Pleo will ship to customers who place these early orders and become available in specialty retail stores in limited locations. Price tag: $249.

Hacking Pleo? UGOBE plans to have a consumer online SDK and a developer SDK — both will be available in 2007. They don't have all the details yet for hacking Pleo as

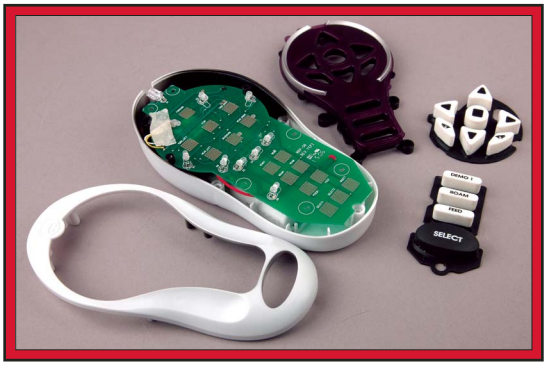
PLEO SEIZES UP!

Later in development, UGOBE was beginning to put on demonstrations of Pleo's capabilities. As they were getting the "motion blending system" up and running to enable Pleo to perform multiple motions one after another, UGOBE was also preparing for another demo.

Working through the night, they were ready for their demonstration. UGOBE CTO John Sosoka had to catch a flight the next day to lead the demonstration. In the middle of the demo, where Pleo had just been going through its motions with flawless perfection, it went into a series of multiple simultaneous motions, overloading its joints and making it appear as if Pleo was having a seizure. Sosoka turned Pleo off quickly, as even as a robot, its movements were so life-like that it looked as if Pleo was in genuine pain.



Front view of Roboreptile and its innards.



Roboreptile's remote control in parts.

Remote control, whole.

those are still being developed.

Roboreptile

Roboreptile is about 80 cm long by 24 cm wide and 15 cm high in four-leg mode. With batteries installed, it weighs in at just under 2-1/2 lbs.

The Roboreptile from WowWee Robotics is powered by five motors: two for the legs and one for the neck, tail, and jaw. By using "high speed resonant locomotion," Roboreptile can walk in a dozen or so different configurations.

Roboreptile uses bi-directional microphones and two IR sensors for its eyes to detect movement and avoid objects. It has a touch sensor to make it sensitive to being touched on its back. Through stereo sound sensors, it can react to sounds in its environment.

A light sensor enables Roboreptile to recognize when its hood has been placed over its eyes to "calm him down." The IR radar also detects, tracks, and moves toward an IR "food beacon" to simulate eating.

Roboreptile employs a custom-built RISC CPU with 128 bytes of RAM, a 12K assembly language codespace, and 1/2 meg of sound ROM.

Roboreptile Capabilities

All this technology enables Roboreptile to do some interesting, life-like things. In its free-roaming mode, it is hungry and angry (missed its night-time feeding, perhaps?). In this state, it explores its environment, avoids obstacles on either four legs or two, and makes a lot of angry, hungry-like noises. It makes a variety of movements that add to the image of angry hungry roaming, just like you might expect to see a dinosaur progress through in a horror flick.

Roboreptile reacts to motion around it as picked up by its IR vision sensor and responds to sounds by chasing whatever is making those sounds.

Another method of calming the Roboreptile is to simulate feeding the robot by using the IR remote control. The IR radar detects the feed signal and the robot turns to face the direction that the signal is coming from. The robot even follows the position of the IR signal with its head.

Roboreptile then jumps down on its four legs and runs to chase the signal, as if running toward the food source. When you release the IR remote button, that's where the robot stops to eat. Having been fed, it will move in a slower, more relaxed fashion, reacting more calmly to sound input.

At this point, you can place the hood over its head and it will calm itself further. Then, it can be picked up and petted on the back near its touch sensor.

Roboreptile — WowWee's fastest and most agile robot to date — exhibits fast anaerobic-like motions that are unique to a robotic creation. It is the fastest walking robot in its size range.

Hacking Roboreptile? For those who might want to hack Roboreptile, all the sensors are in the head and the brain is in its spine. All the inputs and outputs are color-coded and socketed for easy use and access. Motors and gearboxes are double the speed and strength of earlier WowWee robots. This instills quick reaction times into the bot. It also has two new types of gearboxes to keep hackers interested.

Thanks to UGOBE, makers of Pleo, and WowWee Robotics, makers of Roboreptile for their fine consumer robots. SV

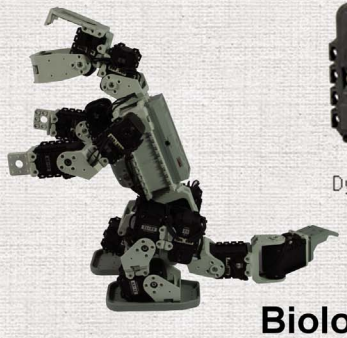
Roboreptile with some of his parts bare for the world to see.

RESOURCES

UGOBE and Pleo
www.UGOBE.com/pleo/index.html

WowWee Robotics and Roboreptile
www.wowwee.com

CrustCrawler Robotics



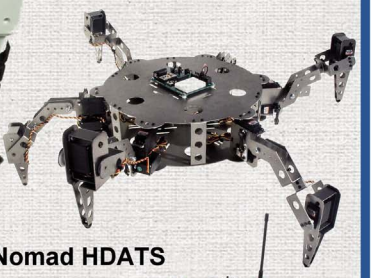
Dynamixel(Servo for Robots)
AX-12 (12Kgf.cm)

ROBOTIS

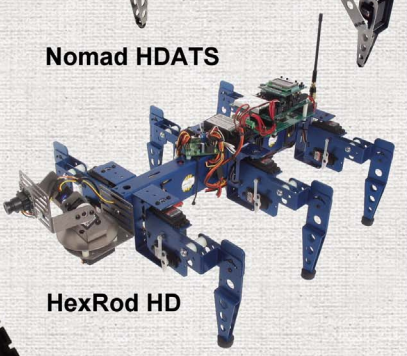
Bioloid Kit
Build over 26 robots
with a single kit
for $899.00!

SG Series
Robotic Arms

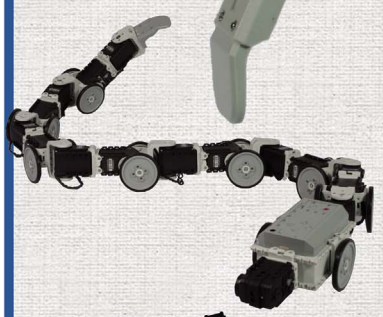
Walkers

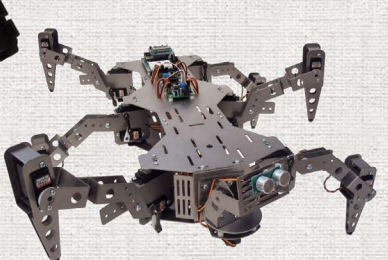


Nomad HDATS



HexRod HD

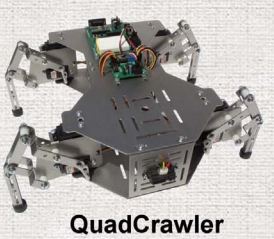




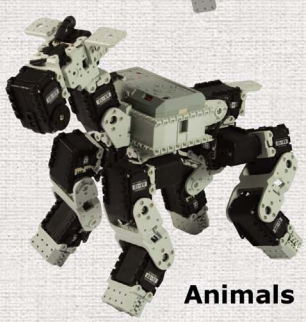
HexCrawler HDATS

Multi-leg

Bipeds



QuadCrawler



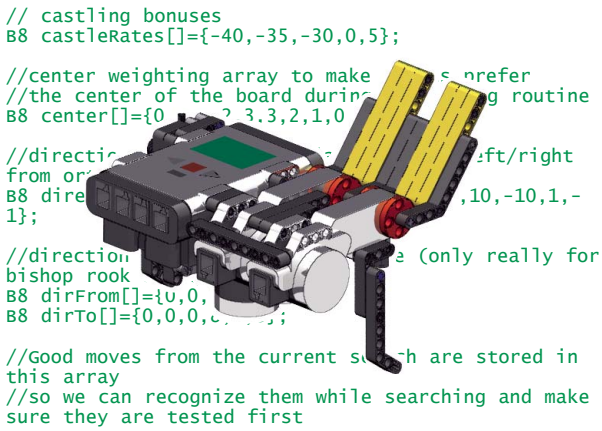
Animals

Wheeled Robots

ROBONOVA-1

HexCrawler

Visit us at www.crustcrawler.com or call 1-(480) 577-5557



LESSONS FROM THE LABORATORY

A bi-monthly column for kids!

NXT Robotics: Remote Control

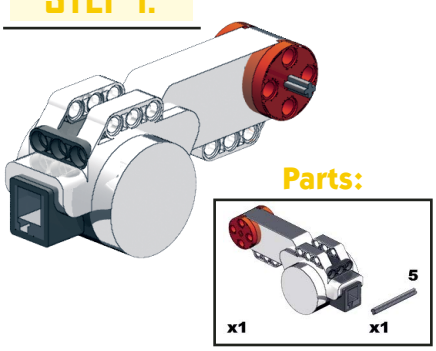
with Brian Davis
by James Isom

This month, we finish off our Brian Davis series with a look at his remote control. The remote uses the Bluetooth capabilities of the NXT brick combined with the rotation of a motor to control the power level on the connected robot.

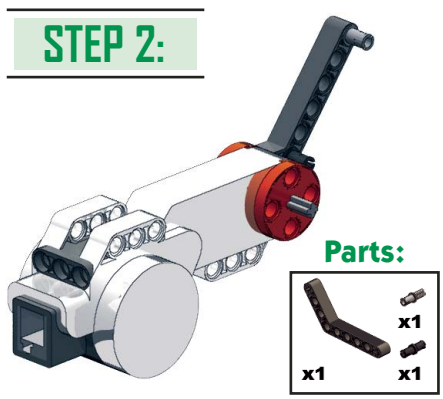
THE CONTROL MODULE

Make this twice.

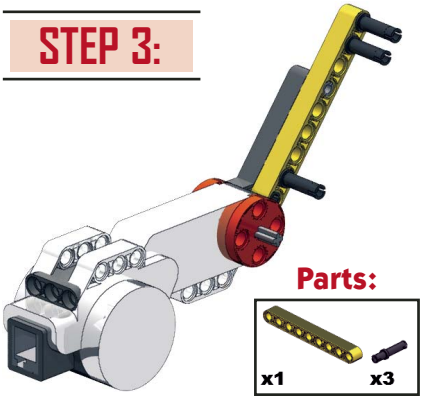
STEP 1:



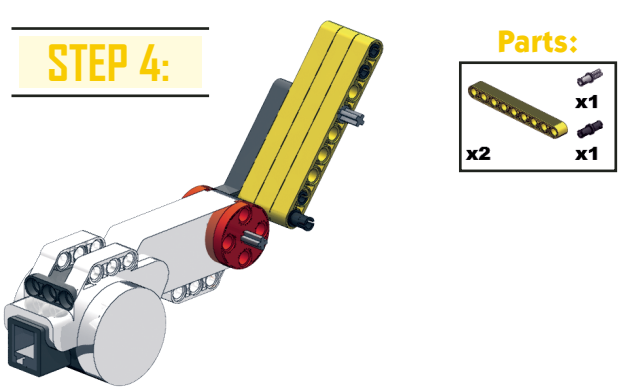
STEP 2:



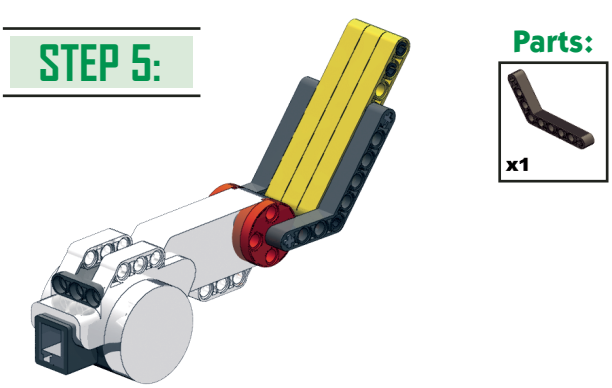
STEP 3:



STEP 4:

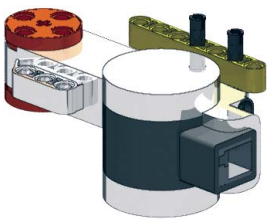


STEP 5:

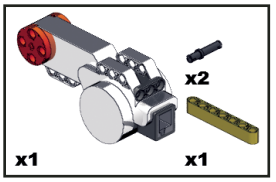


THE FRAME

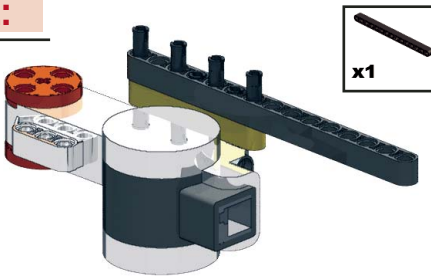
STEP 1:



Parts:

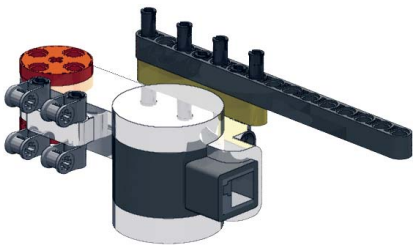


STEP 2:



Parts:

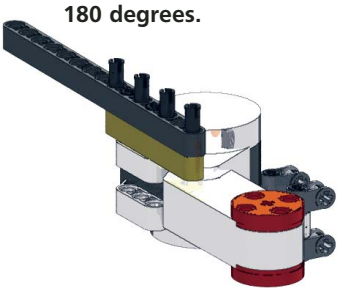
STEP 3:



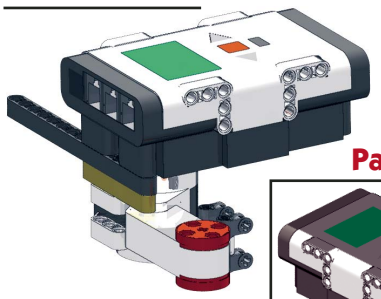
Parts:

STEP 4:

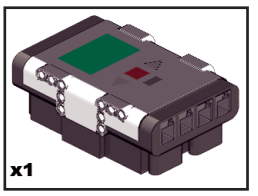
Rotate the assembly 180 degrees.



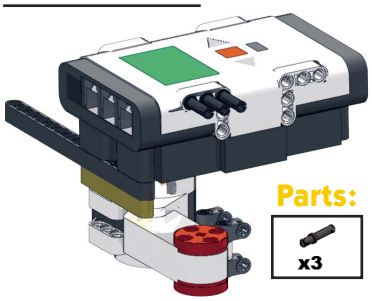
STEP 5:



Parts:

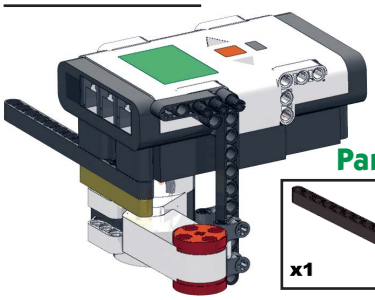


STEP 6:



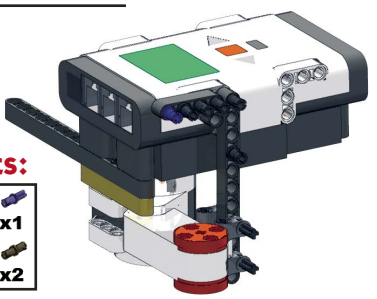
Parts:

STEP 7:



Parts:

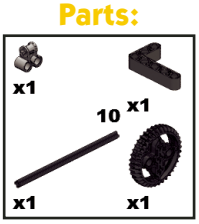
STEP 8:

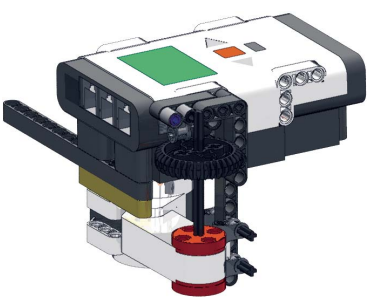


Parts:

STEP 9:

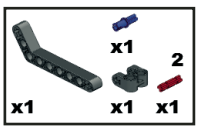
Parts:

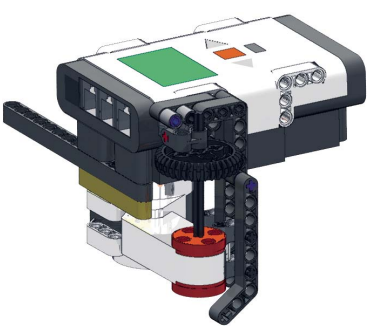




STEP 10:

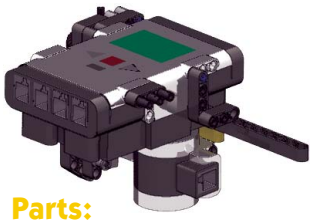
Parts:





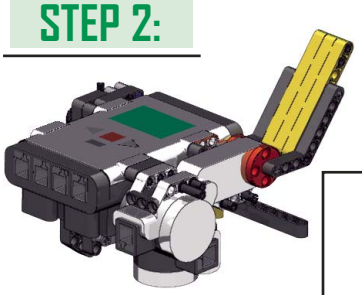
FINAL ASSEMBLY

STEP 1:

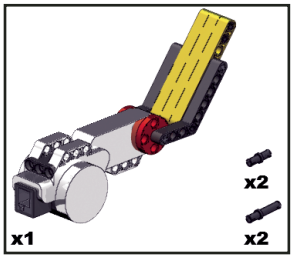


Parts:

STEP 2:

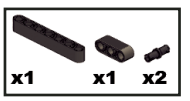


Parts:

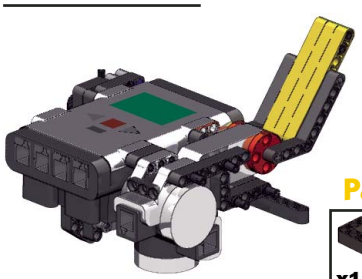


STEP 3:

Parts:



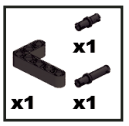
STEP 4:



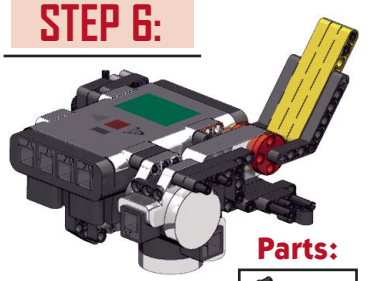
Parts:

STEP 5:

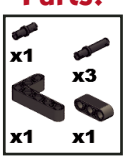
Parts:



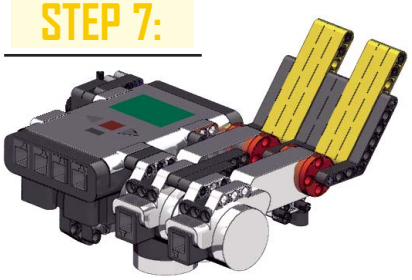
STEP 6:



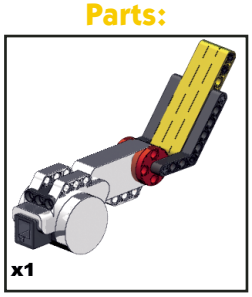
Parts:



STEP 7:

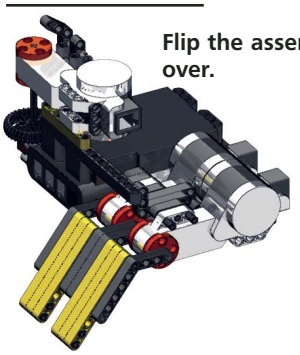


Parts:

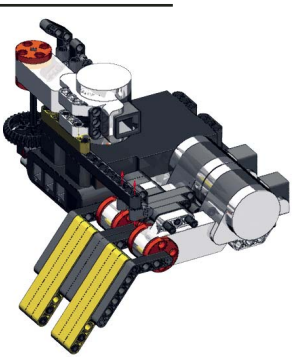


STEP 8:

Flip the assembly over.



STEP 9:



Parts:

Congratulations, you're finished! All that's left is to connect a wire from each motor into one of the output ports on top of the NXT. Output ports B and C should be connected to the two controller paddles. Brian Davis' program for this little beauty and its counterpart to control Jenn Too can be downloaded from my website at **www.LEGOedwest.com**. After you have the programs downloaded to their respective NXTs, simply establish a Bluetooth connection between the two, run the programs, throw the paddles forward, and you're off and running. Have fun! **SV**

Discover Why the Brightest Minds Think Edmund...

NEW! Khet

Strategy at the speed of light.

Combine classic strategy with the physics of lasers and optics to master this game of skill. Egyptian themed pieces with one, two, or no mirrored surfaces move about the game board. Each turn ends by firing a laser beam (built into the board) which bounces from mirror to mirror on a series of pieces. In maneuvering each piece, the aim is to illuminate your opponent's pharaoh and eliminate them from the game. Designed for two players ages nine and up, the game comes with required batteries.

#P30830-59 $44.95

NEW! Cube World

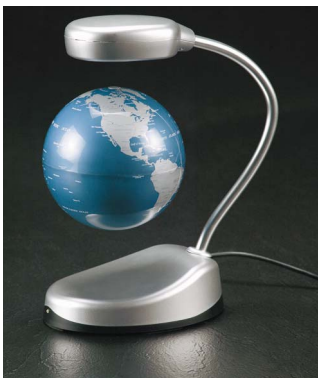
• The coolest new toy of the year!

Discover the distinct personalities and habits of stick figures Slim, Scoop, Dodger, and Whip. These animated stick figures live inside their own colorful cubes, where they play, think, and sleep. Slim uses a stick to do pull ups and martial arts. Dodger bounces a ball. Scoop plays keep away with his pet dog. And, Whip jumps rope. Watch one cube separately and play games with an individual figure or connect the cubes for more interactive fun. Surprisingly, stick figures in connecting cubes will wave to each other and transfer themselves between cubes to fight, play, and even high five. Cubes connect in any horizontal or vertical pattern through magnetic sensors on the cube's sides. Once connected, the possibilities are endless.

Slim/Scoop #P30830-47 $29.95
Whip/Dodger #P30840-23 $29.95

Products That Inspire Science Exploration

For Over 50 Years, Scientifics has been proud to provide you with the latest and best selling, unique science-related items available. From unusual gift ideas, to items you just won't find anywhere else–we have something to make you smile… and think.Whether you are a teacher, scientist, hobbyist – or just love exploring the world around you, our unique collection offers a refreshing alternative.



The IFO 3000 - Magnetic Floating Display

The IFO 3000 takes levitation to new heights. From the impressive floating gap to the aesthetically pleasing frame design, the IFO 3000 will grace any home or office. The uniqueness of the Floating Ideas Magnetic Display makes it the perfect gift. The product is easy to use – the patent pending technology and the spacer training aid that comes with each kit make sure that even a complete novice will be floating straight out of the box. The objects rotate serenely in the magnetic field in either direction. At the flick of a switch and a flick and a gentle push of the object, watch as the object slowly rotates forever. The IFO 3000 is supplied with an Earth Globe floating object as standard. Other floating objects are sold separately.

Globe with base #P30369-82 $99.95
Space Shuttle (without base) #P30369-84 $19.95

Nature's Fire™

• Brilliant Moving Sculpture Created by Physicists

The dynamics of bringing earth, fire and water together are reflected in this amazing kinetic sculpture. Bejeweled lava continuously erupts from the underwater volcano and soft, hypnotic lights enhance the lava's movement in multi-colored patterns. Lighting is provided by LED, so there are no bulbs to replace. It measures 9.7" L x 7.3" W x 2.5" D and includes a 120V or 220V power adapter.

#P31025-28 $34.95

Contact Us Today For A Free Catalog

Over 2,000 Items For Science Gift Giving!

Edmund Scientific's SCIENTIFICS®

Reference Code CO69-WW12

1-800-728-6999 • www.scientificsonline.com

EVENTS CALENDAR

Send updates, new listings, corrections, complaints, and suggestions to: **steve@ncc.com** or FAX **972-404-0269**

We've come to the end of another year of robot competitions, but the dates for 2007 are beginning to roll in. I expect next year to bring more competitions than ever as the interest in robots continues to grow. Enjoy the holidays and perhaps you can use the extra free time to start building robots for next year.

Know of any robot competitions I've missed? Is your local school or robot group planning a contest? Send an email to steve@ncc.com and tell me about it. Be sure to include the date and location of your contest. If you have a website with contest info, send along the URL as well, so we can tell everyone else about it

For last-minute updates and changes, you can always find the most recent version of the Robot Competition FAQ at Robots.net: **http://robots.net/rcfaq.html**

— R. Steven Rainwater

December

1-2 **Texas BEST Competition**
Moody Coliseum, SMU, Dallas, TX
In the Texas BEST Competition, students and corporate sponsors build robots from standardized kits and compete in a challenge that is different each year.
www.texasbest.org

2 **LVBots Challenge**
Las Vegas, NV
Line following, line maze solving, and mini sumo, all for autonomous robots.
www.lvbots.org

8-9 **South's BEST Competition**
Beard-Eaves Memorial Coliseum, Auburn University, Auburn, AL
Regional BEST teams from multiple states compete in this regional championship.
www.southsbest.org

9 **ROBOEXOTICA**
Museumsquartier, Vienna, Austria
A competition for "cocktail robots" that includes events such as serving cocktails, mixing cocktails, bartending conversation, and lighting cigarettes.
www.roboexotica.org/en/acra.htm

9 **Penn State Abington Robo-Hoops**
Penn State Abington, Abington, PA
Autonomous robots pick up foam balls and shoot or dunk them in a basket.
www.ecsel.psu.edu/~avanzato/robots/contests

January 2007

16-19 **Singapore Robotic Games**
Republic of Singapore
Lots of events including wall climbing, pole balancing, Micromouse, sumo, legged robot races, robot soccer, and more.
http://guppy.mpe.nus.edu.sg/srg

20 **Robot Sumo in DC**
Washington, DC
The name says it all. This is a robot sumo contest held in Washington, DC.
www.societyofrobots.com/sumo_robots_in_DC.shtml

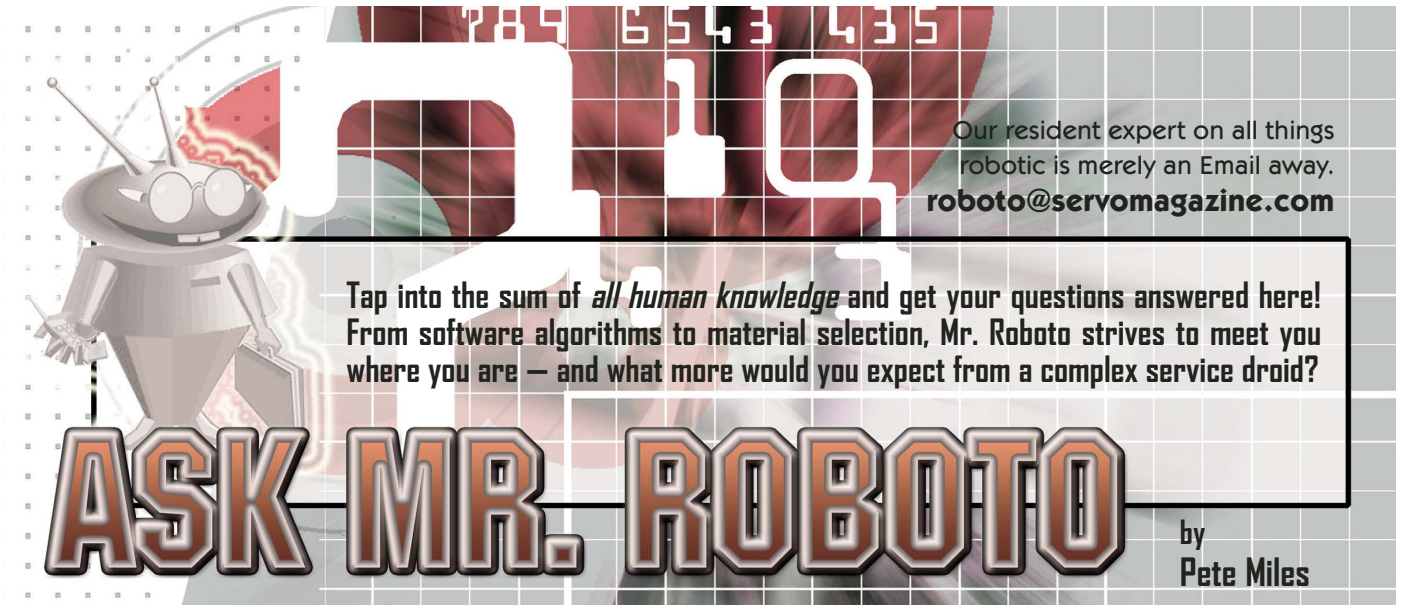
26-28 **Techfest**
Indian Institute of Technology, Bombay, India
Micromouse and two other events with the intriguing names of SNAP and Full Throttle: Afterburn.
www.techfest.org

February

1-4 **Robotix**
IIT Khargpur, West Bengal, India
A national-level competition. Events include Fastrack Manual, Fastrack Auto, and Softandroid.
http://gymkhana.iitkgp.ac.in/robotix

1-4 **Pragyan**
National Institute of Technology, Trichy, India
Events include TrailBlazer and EyeRobot.
www.pragyan.org

26 **APEC Micromouse Contest**
Anaheim, CA
One of the best-known micromouse competitions in the United States. Expect to see some very advanced and fast micromouse robots.
www.apec-conf.org



Our resident expert on all things robotic is merely an Email away.
roboto@servomagazine.com

Tap into the sum of *all human knowledge* and get your questions answered here! From software algorithms to material selection, Mr. Roboto strives to meet you where you are — and what more would you expect from a complex service droid?

ASK MR. ROBOTO

by
Pete Miles

As a recap, in the October 2006 issue Eric posed a question asking why so many sensors output a high signal when it is not detecting anything, and then goes low when it detects something.

Q. I believe the reason that many sensors go low when detecting something is for safety reasons. Many sensors are used to help in controlling and limiting an automated process. It can be really almost anything from machine tools to mixing chemicals to detecting particular proteins. If the task being done has limits that must not be exceeded, then the controller needs to always know that the sensor is operating. It does this by having the sensor complete a circuit. If the circuit is broken by either the limit being detected or a fault in the circuit, the controller stops and the reason for the low state is investigated.

Of course, the above method isn't perfect, but it is simple and cheap and is good enough for many uses because the sensor is never supposed to actuate. For example, limit switches at the extreme travel of a machine tool axis.

Just an FYI in case you're curious.

— Eric R. Snow

A. Eric, you bring up some excellent points here. Safety and reliability need to be considered when wiring sensors into a system. The two things that need to be considered in determining which way to wire a sensor are: how the system will respond when it detects a sensor state change and how the system will respond if the sensor fails. Everyone designs their systems considering the first case, but too many people forget about the second case. They assume the sensor will always work the way they want it to. Unfortunately, in the real world, sensors fail to work as desired. They are either being used in an application they weren't designed for, they malfunction, or there is a wiring problem between the sensor and the main processor.

When working out the wiring logic, one should consider what happens if the sensor fails to work. If the output is normally high, how should the system respond if there is an electrical failure that causes the signal to go low? If the sensor's output is normally low, how should the system respond if it fails to go high when it detects something? As in your limit switch example, if the controller loses communication to a limit switch in a machine tool (i.e., it opens when it is not mechanically triggered), the machine better shut down for safety reasons. But on the flip side, if a sensor's output goes low when it detects something, it will go low if the wiring breaks. If a combat robot is designed to attack when it detects something, a broken sensor would put it into a permanent attack mode which is, in itself, dangerous.

Because there are so many ways a sensor can be wired into a system, most industrial sensors have both normally open and normally closed contacts so the same sensor can be used in either configuration.

Q. I bought a few of the Sharp GP2Y0D340K infrared object detection sensors because of your article in the September '06 issue of *SERVO Magazine*. I managed to get one apart, but I trashed the adjustable lens. I want to use these to detect trains on a model railroad because I know that they will be quite reliable. I built the circuit and it works just fine. Also, the one I destroyed the lens in will still work because I plan to set these in the track. I can leave the other ones alone because the detector will be looking up and will work as sold. I have G gauge so I have plenty of room.

Now, my question — how can I modify the circuit to turn a five-volt relay on and off when there is a train present? I am a connect-the-dots guy when it comes to electronics.

— John L. Deming

A. Sorry to hear that you broke one of the sensors. One would think that if Sharp would advertise that these sensors are adjustable, they wouldn't use such a strong glue. A little pressure is usually all that is needed to break the glue, but some parts may have a lot more glue than others.

Adding a five-volt relay is pretty straightforward. Figure 1 shows how the circuit in the September '06 issue was modified to control a relay. All you need to do is add a second general-purpose NPN transistor, Q2, and a diode, D2. The diode is operating as a flyback diode to protect the transistor from the reverse voltage spike that occurs when the

relay's coil is de-energized. In this circuit, I have moved the location of the LED, D1, to provide a visual indication that the sensor is detecting something. The relay will be energized when the LED lights. The LED and its current limiting resistor, R4, are optional. They can be removed from the circuit and it will still operate as intended. The relay can be either a single pole or a double pole relay, it doesn't really matter. Hope this circuit will help with your model train project.

Q. I have been an avid reader of both *Nuts & Volts* and now *SERVO* for quite some time. I have several types of bots and am currently working on my masterpiece. This one uses a PC/104 main computer talking to various Atmel controllers.

I am trying to find either the software or the tools to create a program that will allow me to send control and status signals via the Internet from any remote browser to the robot. Once the control signals are received by the bot, I would then need to be able to use these in my control software. Status signals would be sent in the reverse direction back to the browser. I'm currently using Visual Basic as the master control software. In addition, I would also like to send streaming video from the bot back to the browser using a standard webcam. Any ideas?

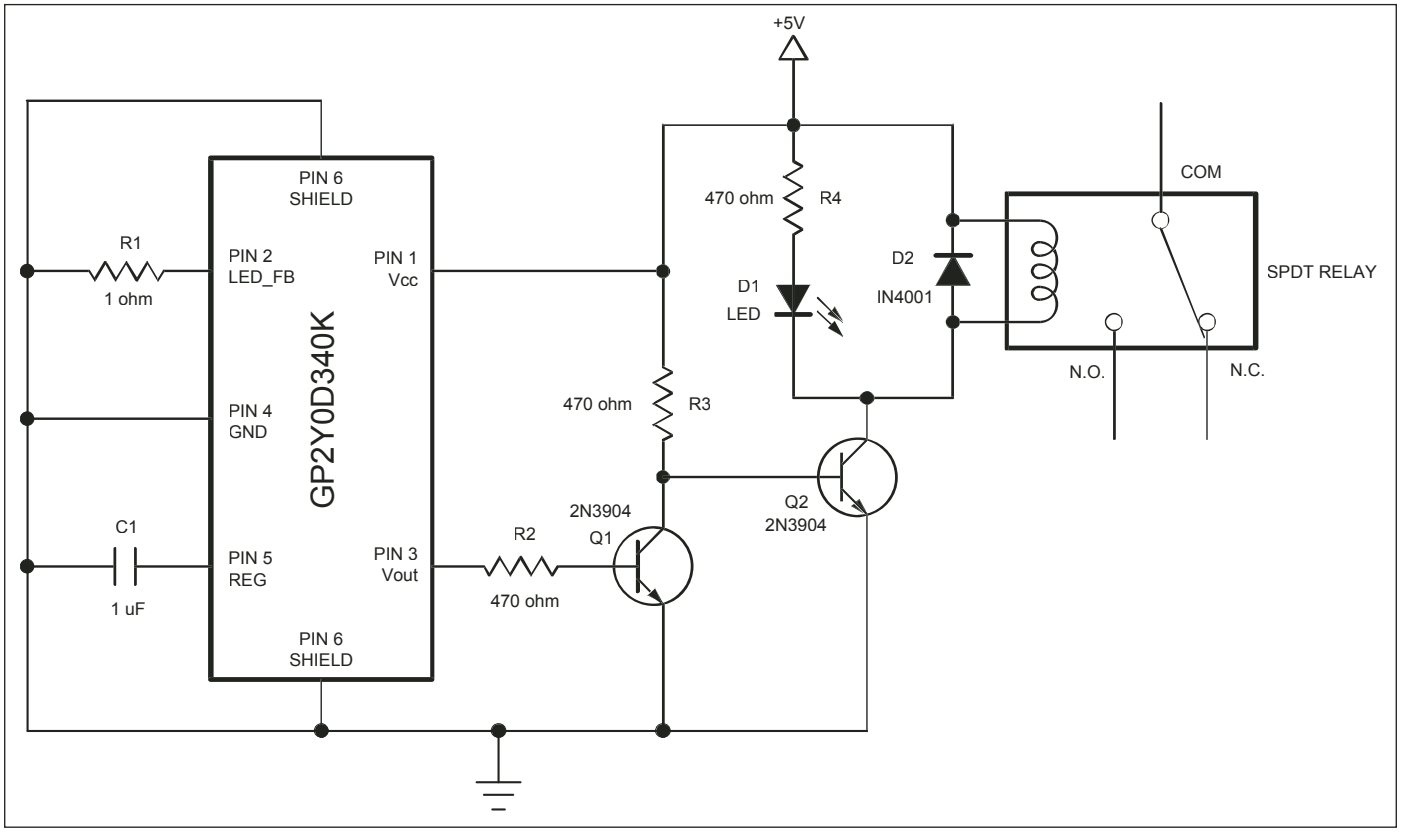
— Don Peterson
Pleasanton, CA

A. This sounds like an exciting and challenging project. The software part of your question is going to be the harder one to answer, so I will address a couple software packages that may fit your needs first. Since you are currently working with Visual Basic, I would suggest that you take a look at the Microsoft Robotics Studio **http://msdn.microsoft.com/robotics/** software, and the ERSP software from Evolution Robotics **www.evolution.com**. Both of these software packages are designed for PC-based robotic applications, and work with Microsoft Visual Studio programming languages such as Visual Basic and C#, and the .Net environments.

You mentioned that you would like to use streaming video with your robot. The software from Evolution Robotics is heavily based on vision recognition and navigation capabilities. They have developed a set of software tools for vision-based applications that work with the Microsoft development software, which is quite impressive, especially with their object recognition capabilities. If vision capability is your main goal, then take a serious look at the Evolution Robotics software.

The Microsoft Robotics Studio is a new product under development by Microsoft, and it is currently available for download free of charge. The Robotics Studio is designed for controlling and communicating with robots, either directly, remotely, or via web-based controls. Several robotics companies such as Kuka Robotics (**www.kuka.com**) and White Box Robotics (**www.whiteboxrobotics.com**) are offering the Robotic Studio as one of the software development options for their products. The Robotics Studio also has demonstration applications for working with LEGO Mindstorms RCX and NXT bricks (**http://mindstorms.lego.com**/), Parallax BoeBot (**www.parallax.com**),

Figure 1. Sharp optical sensor controlled relay.



and the Lynxmotion 6 axis arm (**www.lynxmotion.com**). Many of these robots are controlled via serial or wireless controls.

With your PC104 computer, you are going to need software that can run either Windows CE or Windows XP Embedded (**http://msdn.microsoft.com/embedded/**). I don't know the specifics about your hardware, but the PC104 hardware developed by WinSystems (**www.winsystems.com**) has the software for running both Windows CE and XP Embedded. You will have to check with your hardware manufacturer to see if they support these operating systems and if they are compatible with either the Evolution Robotics software or the Microsoft Robotics Studio software.

Both of these software packages — along with just about any other software package — will work with sending control data to the robot and receive status data from the robot. This can be done via a wired serial communication line or via wireless connection hardware such as the ZigBee wireless modules from Maxstream (**www.maxstream.com**). Since you would like streamless video from your robot using a standard webcam, I would suggest that you take a look at the wireless webcams from D-Link (**www.d-link.com**) and Linksys (**www.linksys.com**).

Hopefully, the information provided here can get you pointed in the right direction in developing your PC-controlled robot. When you get your robot up and running, put an article (or two) together for *SERVO* on what you did to get it working. Many of our readers would love to know how you did it, including myself. **SV**

MOVE IT

DIGITAL ROBOT SERVO
HSR-8498HB

Need to move something? More robots are built with Hitec servos than any other brand. Our dedication to providing custom servos to the robotics marketplace is unparalled. Now, Hitec Robotics is pleased to offer our latest custom robot servo, the HSR-8498HB.

The feature packed HSR-8948HB servo is the muscle behind our popular ROBONOVA-1 robot and can now be used in your hottest robot project.

Build your HSR-8498HB in many different configurations

Features:
- Multi-configurable case components
- Position Data feedback
- Dual Ball Bearings
- Strong Karbonite gear train
- Programmable digital amplifier

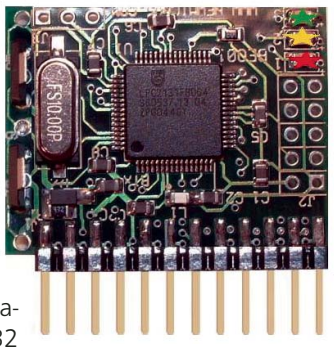
Specifications:
- 180° rotation
- 6V Torque: 103 in/oz
- Weight: 1.94 oz.

HITEC ROBOTICS
www.hitecrobotics.com
Phone (858) 748-6948
12115 Paine St., Poway, CA 92064

NEW PRODUCTS

CONTROLLERS & PROCESSORS

The Tini2131™



The newTini2131™ from New Micros, Inc., comes in the popular Tini pinout format and is based on the Philips ARM LPC2131. The LPC2131 has 32K Flash and 8K RAM. It has on-board regulation, reset circuitry, RS-232 conversion, and three user programmable indicator LEDs. The Tini2131 has 16 of the best I/O pins of the LPC2131 brought out, and separate I²C connections for networking. The 16 shared I/O pins include two 32-bit timers, PWM, and two serials which can be UARTS, I²C, or SPI. There are up to eight channels of 10-bit A/D. This device can be developed in GCC using Eclipse in the same way as the original TiniARM, as well as having leading software tool chains from companies like Keil Software.

Its small size — 1" x 1.3" — allows it to be a tightly integrated solution to robotics, motion, automotive, and industrial control, as well as the capability of being used in networking and data logging applications.

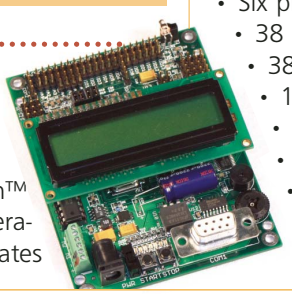
The Tini2131 features a 60 MHz, LPC-2131 32-bit ARM CPU. The popular ARM processor has wide third-party language support, free development tools like the Eclipse development environment using GCC, and a demo version of the Keil compiler, as well. Other languages soon to come from New Micros are IsoMax/Forth and StatiC for the ARM. As an introductory offer, New Micros will include a free Keil demo CD upon request.

A Tini2131 development kit with the serial cable, power supply, and proto-development board is available.

For further information, please contact:

New Micros, Inc.
1601 Chalk Hill Rd.
Dallas, TX 75212
214 • 339 • 2204 Fax: **214 • 339 • 1585**
Website: **www.newmicros.com**

The IntelliBrain™2 Robotics Controller



RigeSoft has just released the IntelliBrain™ 2 robotics controller — its second generation robotics controller — which incorporates the most popular features of the original IntelliBrain robotics controller and the IntelliBrain expansion board on a single circuit board. RidgeSoft has also updated the IntelliBrain-Bot Deluxe educational robot to include the IntelliBrain 2 robotics controller and an ultrasonic range sensor.

The IntelliBrain 2 robotics controller is designed specifically for educational robotics applications. Students program their robots using true Java™ programming — not a Java-like language or other esoteric programming language. Tutorials and a course outline — which are available online — facilitate easy integration into computer science or engineering curriculum.

The RoboJDE™ robotics software development environment, which is included with the IntelliBrain 2 robotics controller, includes dozens of example programs and tutorials covering everything from basic sensing to programming advanced robotic intelligence.

The IntelliBrain 2 robotics controller's design makes it easy to interface with many popular sensors and effectors including hobby servos, DC motors, infrared sensors, sonar sensors, wheel encoders, vision sensors, compasses, GPS devices, speech synthesizers, and many more.

The robotics class library included with RoboJDE provides an easy-to-use, object-oriented programming interface to all of the IntelliBrain 2 controller's features, as well as software support for many sensors and effectors, and an assortment of robotics classes to provide a foundation for programming intelligent robots.

The IntelliBrain 2 robotics controller includes the following features:

- Java programmable
- Two DC motor ports
- Five servo ports
- Seven analog/digital input ports
- 13 digital input/output ports
- Two RS232 serial ports
- Five I²C ports
- Six program-controlled LEDs
- 38 kHz infrared transmitter
- 38 kHz infrared receiver
- 16 x 2 LCD display buzzer thumbwheel
- Two pushbuttons
- Atmel ATmega128 CPU
- 132 KB RAM
- 128 KB Flash
- 4 KB EEPROM

The IntelliBrain-Bot Deluxe educational robot includes:

- IntelliBrain 2 robotics controller
- Two servo motors
- Two wheel encoder sensors
- Two line sensors
- Two infrared range sensors
- Ultrasonic range sensor
- Chassis, wheels, and required hardware

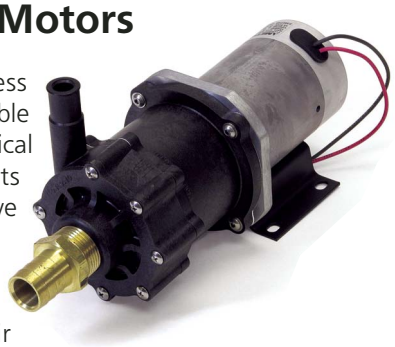
The IntelliBrain-Bot Deluxe educational robot kit can be purchased either assembled or unassembled.

For further information, please contact:

RidgeSoft, LLC
PO Box 482
Pleasanton, CA 94566
Email: **info@ridgesoft.com**
Website: **www.ridgesoft.com**

MOTORS

Brushless DC Motors



DurA-Tek® brushless DC motors available from AMETEK® Technical & Industrial Products feature integrated drive electronics enabling enhanced motor controllability in a smaller, lighter package. Their 3.0-inch outside motor diameter and standard two-wire electrical hookup further allow them to replace similarly sized brush-commutated DC motors while delivering relatively higher performance and extended service life.

These motors ideally suit demanding high-duty cycle applications for equipment used in the transportation industry. Applications expand into HVAC, chemical, mining, medical/biotech, data storage, semiconductor processing, automation, and other industries that can benefit from compact and rugged motor construction. For harsh environments, these motors have been designed to resist hot-water spray, rain, humidity, salt, fog, shock, and vibration.

"Smart" onboard motor controls and advanced electronic design deliver key features, including multi-speed operation, over-current control, locked-rotor protection, reverse polarity protection, transient over-voltage protection, and over-temperature shutoff.

These 12V/24V brushless DC motors can achieve continuous torque from 19.7 oz.-in. to 55 oz.-in. and speeds from 2,927 RPM to 4,400 RPM, depending on the model. Custom products can be engineered.

For further information, please contact:

AMETEK Technical & Industrial Products
627 Lake St.
Kent, OH 44240
330•673•3452 Fax: **330•678•8227**
Website: **www.ametektip.com**

RADIO CONTROL

Control Your R/C Vehicle or Robot From Your PC

Endurance R/C offers an interface system called the PCTx. This device allows a user to control a radio control vehicle or robot via a PC. The PCTx was developed with the intention of providing users with a low cost means to achieve wireless control via a PC. Also, by utilizing hobby transmitters, no modifications to the vehicle or robot are necessary in order to achieve PC control.

The software required to operate the PCTx has been openly released on their website. Endurance R/C feels this allows for a greater level of innovation and also allows many new applications to be developed around the system.

The PCTx requires a radio transmitter with a buddy box/trainer port and a Windows PC with USB port. A universal version for pistol grip style transmitters is currently in the works. PCTx technical specs include:

- Supports up to an eight channel radio system
- Buddy box/trainer port enabled radio required
- Pulses refreshed at 50 Hz
- Independent servo control on all channels
- C++ software API available, VB coming soon
- Upgradeable firmware
- USB 2.0 compatible

For further information, please contact:

Endurance R/C
Website: **www.endurance-rc.com**

Show Us What You've Got!

Is your product innovative, less expensive, more functional, or just plain cool? If you have a new product that you would like us to run in our *New Products* section, please email a short description (300-500 words) and a photo of your product to:

newproducts@servomagazine.com

Halloween Robot Terror 2006

The first ever Halloween Robot Terror is over and everyone had tons of fun. There were two brand new teams competing in their first ever event. Team Bad Bots with their bot Black Wedge is from Palo Alto, CA. The second new team is un-named as of yet with their bot Screamer, a modified BB toy. Also Team Misfit has a new builder/driver, Dan, driving a flea weight named Atom Bomb. Welcome to the sport guys! Hope you have as much fun as the rest of us. The costume contest was a great success and had nine bots compete. I kept hearing a lot of builders saying "next year I'm going to do ..." so I will be doing the Halloween Robot Terror again next year. The winners of the Bot Costume contest are:

- First Place — Stumpy from Team DMV
- Second Place — Front Kick from Team Kick-Me
- Third Place — Scream (the brand new team that's un-named for now)

Photos are posted on the CIB website at **www.calbugs.com**.

In the Flea Weights, there were only two competitors, so I fought them for the best of two out of three:

- First Place — Change of Heart from Team Misfit driven by Kevin
- Second Place — Atom Bomb from Team Misfit driven by Dan

In the Ant Weights, I had seven bots competing:

- First Place — Fire Eagle from Team Misfit driven by Kevin
- Second Place — Stumpy from Team DMV driven by David W.
- Third Place — Pooky from Team ICE driven by David L.

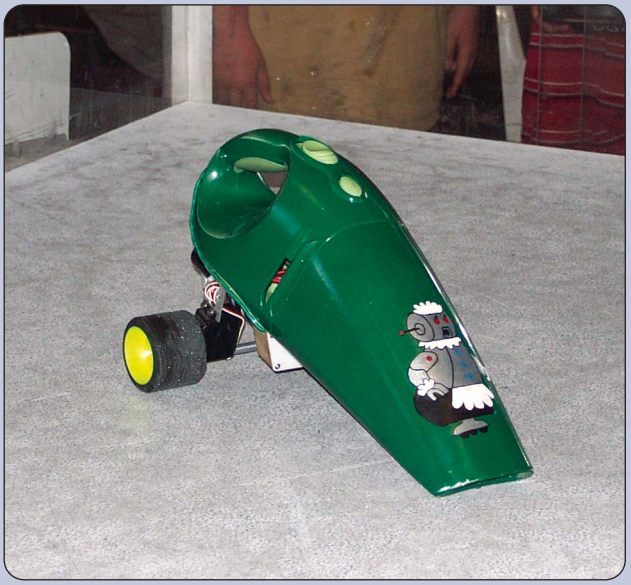
In the Beetle Weights, I had three robots competing, so I ran them round robin.

- First Place — Toe Poke from Team Kick-Me driven by Hugh
- Second Place — Unknown Avenger from Team ICE driven by David L.
- Third Place — Itsa from Team Bad Bot driven by Mike

Dave Wiley
Bot Gauntlet Baron

Continued on Page 84

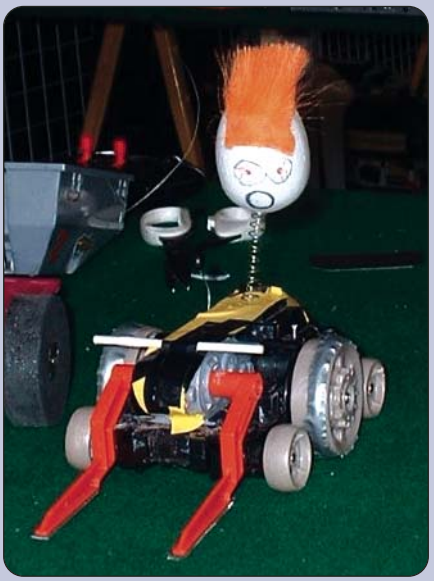
Costume Contest Winners.



Rosie has gotten a LOT of use at several events by people from the audience and it's easy to get a line of people wanting a turn to drive her around. This helps keep the fighting surface very clean.

This is team DMV Ant Weight Stumpy in costume. The eyes light up and the head turns left and right. Also, the handle bars turn when Stumpy turns left and right. This costume took first place.

Third place costume winner Screamer driven by Gabriel. His team is so new its not even named yet.



COMBAT ZONE

Featured This Month

Participation

26 *Introduction to Arenas* by Kevin Berry

28 *Weapon Safety* by Brian Benson

Feature

28 *Voyagers Robotics* by Tim Wolter

Events

30 *Results — Sep. 12 - Oct. 13*

32 *Upcoming — Dec. and Jan.*

Technical Knowledge

31 *Radio Systems* by Leonard G. Ginn, Teampyramid

Product Review

32 *DuraTrax IntelliPeak AC/DC Mini Pulse Charger* by Kevin Berry

PARTICIPATION

Introduction to Arenas

● by Kevin Berry

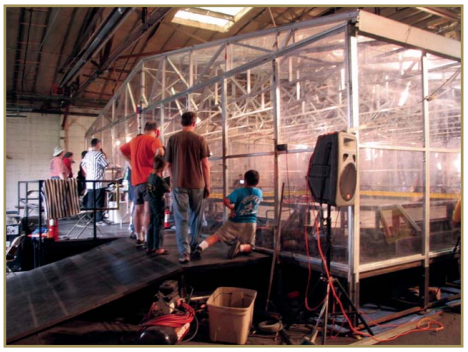
The subject of arena design is always good for a discussion, debate, and (usually) a disagreement. One thing no one disagrees on, however, is that the arena is the first and last line of safety between hard, sharp bots and soft, squishy people. Despite a few close calls over the years, the arenas built and used in Combat Robotics have a sterling record of protecting people from harm.

The first rule of arena design is to assume it must contain the next class bigger than you're planning to fight in it. So, if you're building for the 30 pound "featherweight" class, your arena should be able to withstand the weapon loadout of a 60 pound "lightweight." An insect arena for ants and beetles should be able to contain a Mantis or even Hobbyweight weapon. This is above and beyond whatever safety factors are built into the basic design.

Most arenas are built from a combination of polycarbonate ("Lexan" is a commonly used commercial term for this), along with steel, aluminum, and wood. Insect class arenas need at least 1/4" poly, and big bot boxes range from 1/2" to 1"

The Battle Beach arena being set up. Note the steel floor, kick plates, and suspended wooden ceiling.

Inside the Battle Beach box. Back walls are steel; railroad ties keep bots off the polycarbonate. Overhead lights help with visibility. Photo is from perspective of drivers behind protective window.



Combots arena showing drivers stations. Roof structure is built with trusses. Spanning large arenas is sometimes a difficult design feature.

Inside the Combots arena, showing bumpers designed to keep bots away from the walls. Spinning weapons just a few feet from spectators require lots of thought into a safe design.

depending on design and weight class. Insect arenas run from a tight 4′ x 4′ to a roomy 8′ x 8′ in size, while large bot boxes are 12′ x 12′, up to 32′ square. They tend to run in 4′ increments due to the availability of polycarbonate in 4′ x 8′ or 4′ x 12′ sizes. Ceiling heights are generally 4′ for insect bots, and 8′ in large boxes, for the same reason.

Keeping the bots off the walls is a key design feature, and most arenas incorporate either fixed or "loose" barriers such as "I" beams or railroad ties to do this. Ceilings are often plywood, sometimes covered with moving blankets or other debris absorbing media. Doors can be a weak point, since they must be opened and closed twice for each fight, but they still need to provide the same level of protection as the rest of the box.

Setup, teardown, and transportation also figure into the design equation. Insect boxes can (usually) be set up in a couple of hours, while large arenas might take two days.

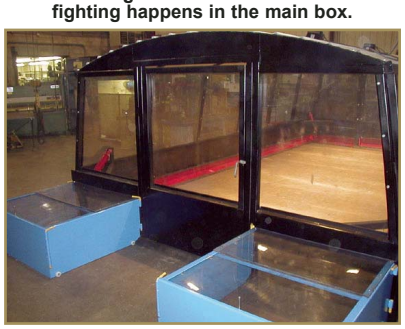
Another design factor is maintenance. After almost every event, some sheets of poly need to be replaced. Also, floors take a beating. Even during events, parts of the arena may need to be repaired or replaced. Other factors include lighting, ventilation (especially if internal combustion engine (ICE) bots are allowed), audience visibility, driver visibility, and what type of surface the arena will be used on. Setting up a box on a concrete slab is a much easier task than on dirt and grass!

This brief overview isn't intended to take the place of a thorough design discussion on current or new arenas. However, as a recurring topic on forums and bulletin boards, it's obviously an important one. Maybe it's time each organization in the sport took a look at their arenas, along with the "arms race" of increasingly powerful weapons, and ask themselves some pointed safety questions, like "is our box still safe enough?" **SV**

"Toad Tank" showing roof support and steel box beam bumpers.

The new SECR Florida insect box being assembled. This arena was built using 80/20 aluminum products, 1/4" polycarbonate, and a plywood floor with sacrificial hardboard overlay.

Team Whyachi's arena has small boxes to allow testing the next match's bots while fighting happens in the main box.



The WarBots arena has a classic setup, with drivers platforms and a ramp in between.

The TC Mechwars arena features a clear roof to increase visibility in large venues.

Weapon Safety

● by Brian Benson

Weapon safety locks are one of the best ways to stay safe around combat robots. They provide the last element of protection between you and the robot when precautions fail or mistakes occur. The purpose of a weapon lock is to prevent the motion of any weapon with the capability of dangerous movement.

Three Basic Criteria

A weapon lock should meet three basic criteria. It must: prevent dangerous movement of the weapon; be clearly visible in shape, size, and color; and be able to be removed and inserted quickly and easily. The removal or insertion of the weapon lock is the most important criteria aside from effectiveness and is the last action that should be performed before leaving the arena and the first when reentering the arena. Every second that is wasted while trying to remove or insert the lock, leaving the weapon to move freely, is one more second where a fatal mistake or error could occur.

When it's Necessary

A weapon lock is necessary every time a robot is turned on; regardless of the situation. Why? Mistakes can and do happen and when it's yours or other people's safety on the line, it's worth the extra effort.

For example, when testing the lifting arms on my 30-lber, I checked the radio settings and turned the robot on. The arms instantly activated as a result of plugging the motor into the wrong receiver port, causing dangerous unexpected movement. Remember, always use your safety lock!

Techniques/Methods

While every robot is different, the most effective and easiest weapon lock is a simple pin or bar. Spinners often use a hole that is designed into the spinning weapon along with a matching hole in the frame for a pin to drop into. Another option is to put two holes on either side of the weapon so that inserted pins will prevent motion in both directions. An excellent example of a weapon lock is shown in Figure 1. For every case, there is a different best way to secure your weapon. The task then is to simply use common sense and find that best way that satisfies the three basic criteria.

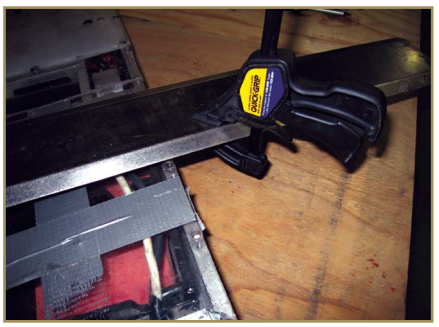
What to Avoid

Many builders leave the weapon lock as a last thought, something to figure out after all the "important" stuff is correct. However, this often results in locks that do not function safely. Improper examples of weapon locks include attachments that require they be clamped, bolted on, or screwed in before they become effective. A perfect example of what not to do is shown in Figures 2 and 3. **SV**

FIGURE 1. Ziggy of Team CM Robotics uses a steel bar to secure their system in the extended and retracted (not shown) position, preventing any dangerous motion. The weapon can output a force of 14,100 lbs, so they use a bar capable of handling 80,000 lbs of force.

FIGURE 2. This is a spinning weapon being held only by a clamp — exactly what should *not* be done.

FIGURE 3. Even small bots need thought put into weapon restraints. This Mantisweight restraint is definitely not safe!



VOYAGERS ROBOTICS

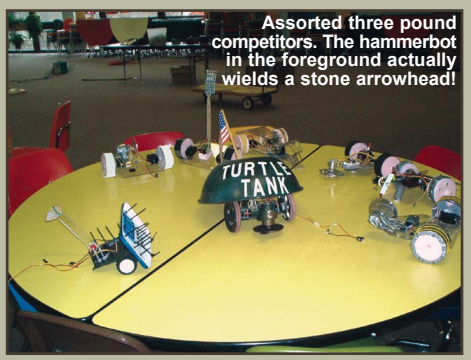
● by Tim Wolter

When my son Karl was still in fifth grade, we started building combat robots together. It just seemed like a natural progression for a kid whose first word was "broken," and who got a sledge hammer for his third birthday.

We had a lot of fun with this, so when he was in middle school, I went to the organizers of the local after-school program with an offer. I said, "Hey, I am willing to teach a class where we let hyperactive sixth graders build heavily-armed remote control juggernauts."

They thought that was an absolutely appalling notion, but after some discussion, we compromised on a smaller scale venture that has been running with success for the last five years.

We take classes of 24 kids at a time and have them each build an R/C combat robot of one or three pound size. The "final exam" for each class is an all-out tournament where the students employ their engineering and driving skills to try and reduce their opponents to smoking rubble.



Assorted three pound competitors. The hammerbot in the foreground actually wields a stone arrowhead!

Frantic battery charging.

And we do this with a near zero budget and a high level of safety. To accomplish this, we "freeze" the technology level so that it is affordable, and so that all students compete with similar equipment.

The most basic machine utilizes two Hitec servos which the kids "hack" for 360 degree excursion and couple to drive wheels. These can be hooked to a standard R/C receiver and a six-volt NiCad or NiMH battery to make a basic "pusher" or wedge robot.

Most students want to build something just a bit fancier. With a bit of soldering, a four-wheeled machine can be built by coupling two servos per side. And active weapons can be added using 9.6 volt R/C car drive packs connected to motors driving spinning bars or discs. Weapon control is via a micro servo that closes a circuit as a simple mechanical relay, or with low-cost airplane electronic speed controllers.

I make a point of encouraging far out designs. Other than the weight classes, the only rules are:

1. No flame throwers.
2. No hand grenades.
3. No live animals.

So, we have had robots built from old Nintendo controllers, sponges, wood, and mixing bowls. They have featured armament ranging from six inch circular saw blades, to a hammer bot wielding a stone arrowhead, and more than a few machines that go into battle equipped with little more than a good paint job and ill founded confidence.

The cost of the program is low; each student pays a $20 class fee, which we waive for financial need. Hitec servos — usually HS-311 or HS-325HDs — run about $8-$10 each. An excellent source for these can be found at **www.servocity.com**. I also usually have a few donated machines or wreckage left over from previous class sessions for kids who want to try a four-wheel version. I have connections that provide me with Lexan scrap that comes in handy. And for the electronics, I have been gradually accumulating transmitters, receivers and batteries off of eBay, from donations, and from various low cost sources, such as **www.batteryspace.com**. Tom's RC (**www.tti-us.com**) has some good electronics, as does the Robot Marketplace (**www.robotmarketplace.com**). As the world of combat robotics is heavily populated with, well, techno-nerds, it is not surprising that the Internet is the necessary glue that links the community together.

There are also many useful components to be found for free. I generally have one session I call "junk day" where kids are encouraged to bring power tools, toys, small appliances, and such to be cannibalized. I hope that these are non-functional items, and that parental permission was granted! It has become necessary to specify no VCRs; with the advent of the DVD player, there are piles of these sitting around, and they yield few useful components.

To do a class of this sort, it is necessary to break the group up into two groups of 12 students each. That is about the maximum number of adolescent goofballs that can be contained in one place and watched while they use tools. Each class gets about eight sessions of roughly 1.5 hours of build time. Interestingly, while my classes have been about 95% male, the girls who do participate always build outstanding machines.

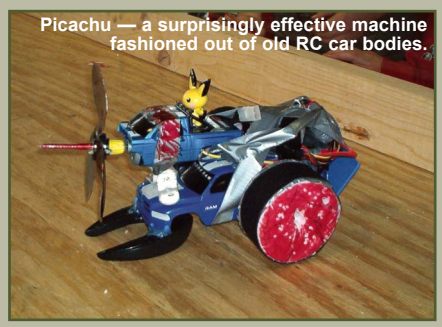
So far, the middle school has been supportive. We get a tech ed classroom to work in, a certain amount of storage space, and a chance to do our end-of-class tournament at the school on a Saturday afternoon. Teacher participation has been helpful but intermittent, so the show is usually run by myself, my son, and various parent volunteers.

Since the school shop is usually not available, we bring our own tools, which include a soldering iron, cordless drill, small drill press, and a sheet metal shear for cutting Lexan. Each kid brings a shoebox to contain his work in progress.

Our school has a strict no weapons policy that probably bans everything beyond plastic forks, so I have kids bring any dubious items directly to the class storage area. So far, nobody has had any problems on the bus to school.

The goal of the class is to have fun, but along the way, we do manage to teach the kids a bit about design, use of materials, radio frequencies, wiring circuits, and use of some basic tools. With close attention to safety glasses and a ban

Emergency repairs.


Picachu — a surprisingly effective machine fashioned out of old RC car bodies.

A possible world record for percentage composition of duct tape.

on unsupervised weapon activation, we have had no injuries worth mentioning. We apply about one band aid a year, usually to some part of me.

The tournament is a major undertaking. I have a 6' x 6' Lexan enclosed arena with a plywood floor. Arena hazards vary with my whims, but usually include a saw and a grinder, along with a trap door. Arena hazards are run off a control board with audience volunteers, which are never in short supply.

We are not yet at the point where we have 24 sets of electronics, so hard-working volunteers continuously swap receivers and batteries from one machine to another using Velcro and plenty of duct tape. Local R/C enthusiasts and combat builders from the Twin Cities Mechwars group have always been there to pitch in.

We usually weld up trophies from whatever junk is lying around the workshop, and award them for first through third in each of the weight classes. There is also a special award for best design, by which I mean most innovative, not necessarily most successful. The highest award of all is the coveted "Golden Dumpster," given for most effective use of junk materials. We live in a time when hands-on tinkering is less common than it once was, and I believe in encouraging kids to scavenge and experiment as much as possible.

With the generally declining cost of electronic components, the class is in some ways getting easier to run over time. But looking ahead I suspect that I will at some point give in to the temptation to have the class start building more advanced machines, perhaps with antweight controllers such as the Scorpion, or even 12 to 15 pound machines for one of the several competitions that run in our area. SV

EVENTS

RESULTS — September 12 - October 13

Fall W.H.R.E '06 was held 9/16/2006 in Dorchester, WI. Results are as follows:

WHRE
Whyachi House of Robotic Entertainment

• *Heavyweight* — 1st: "Ty," plow, Bobbing For French Fries; 2nd: "Brick," wedge, Whyachi.

• *Middleweight* — 1st: "Falcon," drum, Whyachi (RFL Ranked #1); 2nd: "Maxo," spinner, RoboRednecks.

• *Lightweight* — 1st: "Goosfraba," wedge/flamethrower, Killerbotics.

• *Featherweight* — 1st: "Pyromancer," wedge/flamethrower, Killerbotics.

• *Hobbyweight* — 1st: "Ricochet," wedge, Whyachi; 2nd: "Shroom of Doom," wedge, Delta Strike Force 2001.

• *Beetleweight* — 1st: "Firefly," wedge, Booyah; 2nd: "Jeepy Jeep"; 3rd: "3A," spinner, Whyachi; 4th: "Celebrity Lunchbox," ram, LovBots.

• *Antweight* — 1st: "Underwhere?!," spinner, Hazardous Robotics; 2nd: "ANTI," spinner, 564 Robotics; 3rd: "Nano Falcon," drum, Whyachi; 4th: "Wykydtron," spinner, Delta Strike Force 2001.

• *Fairyweight* — 1st: "Kankle Killer," spinner, Whyachi; 2nd: "Destroyer of Grass."

Robothon Robot Combat 2006 was held 9/30/2006 in Seattle, WA. Results are as follows:

• *Hobbyweight* — 1st "Shear Terror," spinner, Sparkle Motion; 2nd: "Fiasco," spinner, Velocity; 3rd: "Scratch," clamp, Gausswave.

• *Beetleweight* — 1st: "Hurty Gurtie," drum, Death By Monkeys; 2nd: "Creepy Crawler," wedge, X-Bots; 3rd: "Altitude," spinner, Velocity.

Marin Ant Wars VI was held 9/30/2006 in Tiburon, CA. Results are as follows:

• *Antweight* — 1st: "The Bomb," drum, Misfit; 2nd: "Emsee Fry Pants," drum, Burntpopcorn.net; 3rd: "MC Pee Pants," drum, Fatcats; 4th: "Unblinking Eye," spinner, Hammer Bros.

• *Fairyweight* — 1st: "Microdrive," lifter, Misfit; 2nd: "Hulk Hogan," clamp, Fatcats; 3rd: "Crisp," flamethrower, Offbeat; 4th: "Catch 22," drum, Hammer Bros. SV

TECHNICAL KNOWLEDGE

Radio Systems

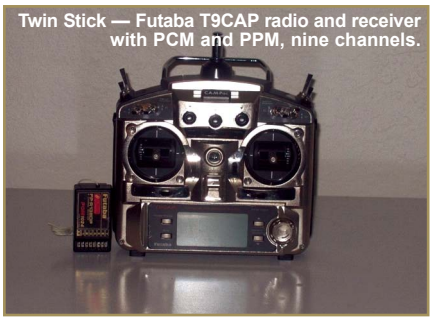
● by Leonard L. Ginn, Teampyramid

So, you are going to build a combat robot. One of the most important things to think about is the type of radio system that you are going to need. If you are building a large bot class (more than six lbs.), you will need a PCM (pulse code modulation) type radio system or better. For the small Insect robot class (six lbs. or less), you may use a PPM (pulse proportional modulation), or you may get by with a toy radio system. This may also depend on the rules of the events you plan to attend with your bot. So check before you invest. The radio system has two parts: the transmitter, which you hold in your hands; and the receiver, which goes in the bot. The radio can be a trigger style or twin stick.

Most electronic speed controls for combat robots only have two or three channels, with sometimes a fourth "invert" function. The twin sticks radio's have two or more channels (up to 6-8), depending on the type. With the twin stick system, you can drive the robot with tank turn steering. Typically, radio systems can come in five different frequencies. They are: 27, 49, 50, 72, and 75 MHz.

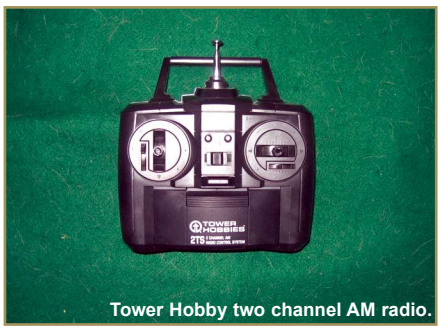
The new radio system on the block now is the 2.4 GHz. This unit, when first activated, searches 80 channels in the 2.4 GHz band. When the radio locates the best clear channel, it will lock on to that channel. The 2.4 GHz frequencies are so much higher than those of conventional channels that your radio won't even recognize the existence of the other bands.

Trigger Style — Futaba Magnum Sport FP-T2PB, two channel.


Twin Stick — Futaba T9CAP radio and receiver with PCM and PPM, nine channels.

The 50 MHz radio is a problem because it requires you to have an amateur radio license from the FCC. The 72 MHz systems are for aircraft use only, and are not allowed for combat robots. The toy radio systems are 27 MHz and 49 MHz. With 27 MHz, you may have a choice of different channels from A1 to A6, and 49 MHz has only one usable channel. There are also many higher quality 27 MHz systems used for R/C cars and boats. The 75 MHz system uses channels 61 through 90. You do not need a different radio to change channels; you can simply change the crystal in the transmitter and receiver. You will, however, need to check with the radio manufacturer on this.

The 75 MHz radio system comes in three different bands: AM (amplitude modulation), FM (frequency modulation) and FM PCM. The AM radio range is extreme-


Tower Hobby two channel AM radio.

Spektum DX6 six-channel DSM 2.4 GHz system from Robot MarketPlace.

ly limited and has a lot of interference. It should never be used on a bot that has a weapon.

FM radios are better than AM — your radio will have less interference. The middle range FM radios are PPM which is an analog system. The better FM radios have what is called PCM which is a digital system. The FM-PPM radio can also have PCM. The difference between the two is in how the signal is encoded. PCM signals are encoded digitally and give a higher degree of immunity from noise than PPM. The best way to find out the type of radio that would work best for you is to talk to other combat robot builders. With a good radio choice in the beginning, you will definitely be able to use it in more than one robot. SV

PRODUCT REVIEW — DuraTrax IntelliPeak AC/DC Mini Pulse Charger

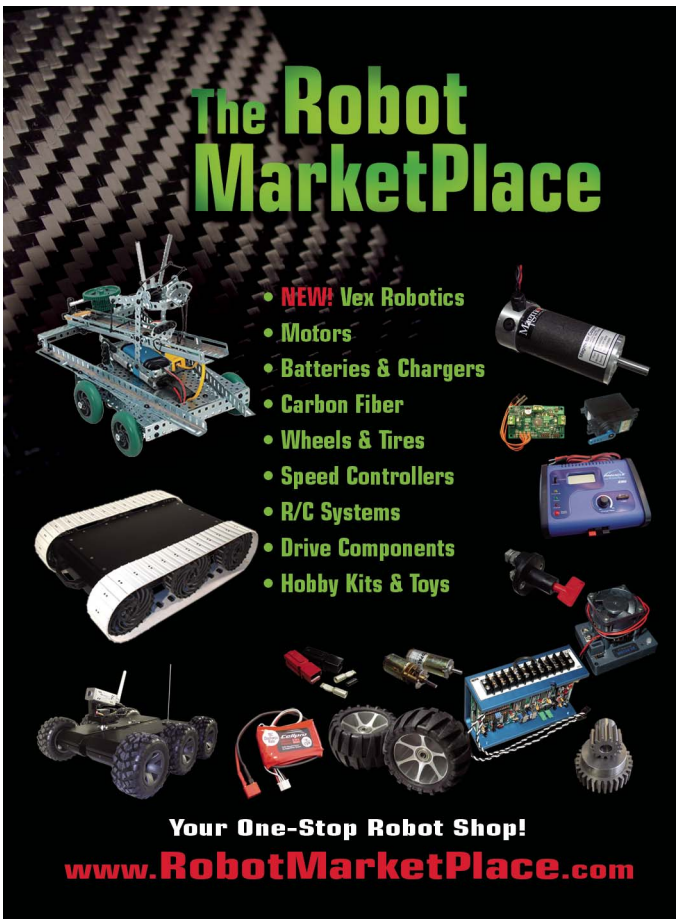
● by Kevin Berry

This was the first charger I bought after I got into the sport, at the recommendation of a mentor who came up through the R/C car racing side of things. Priced new at $64.99, I picked it up on eBay for about $35. I've found it to be a very dependable, basic charger that's handled my NiMH and NiCad charging needs very well.

The model DTX4110 Mini Pulse charger comes with a handy, detachable, 12V power supply that provides seven amps. This has come in handy, both as a quickie bench supply, and the provided alligator clips from the charger itself lets me charge packs from a small sealed lead acid battery in the van on the way to events.

The Mini Pulse can charge up to three amps for up to eight NiMH cells. NiCads can be charged at up to 4.5 amps. There is a two-amp constant discharge function which, to be honest, I've never used since my packs get discharged in driving practice or competition.

I can recommend this as a good starter setup if you can buy it used or on eBay. If you are buying new, there are many options in this price range, so keep in mind that a lot of comparison shopping might be needed. SV



EVENTS

UPCOMING — December and January

ComBots Cup 2007 — This event will take place on 1/14-15/2007 in Oakland, CA. There is a $10,000 Heavyweight prize and a $3,000 Middleweight prize. Visit **www.robogames.net** for more details.

Bay Area Robot Fights — This event will take place on 1/27/2007 in St. Petersburg, FL. It is the fourth event in this annual series — a conventional insect battle run by some very unconventional people. Fun for the whole family. This event data is tentative at time of publication. SV

CollaborativeROBOTICS

The normal course of human social and cognitive development begins with total dependence at birth and progresses to the semi-autonomy of adolescence. Most successful adults advance to a stage of collaborative interdependence that acknowledges the benefits of working with others toward a common goal. This social evolution is reflected in robotics, where there is increased interest in developing robots that can not only work with each other, but work collaboratively with humans.

by Bryan Bergeron

The promise of collaborative robotics is a new breed of machines that are freed from the role of obedient automaton or tele-robot in need of constant supervision. Instead, they can engage in a dialogue with humans, ask and answer questions, and resolve differences in order to achieve shared goals. As a result, humans are free to focus on more important tasks, only occasionally interrupted by robots in need of assistance to compensate for their limited autonomy.

In practice, creating robots capable of collaborating with humans is no mean feat. For example, not only must a collaborative robot have the ability to act autonomously, but it must be able to modulate the level of autonomy to suit the situation. Furthermore, there is the human element to consider. It will take time for people to accept the idea of working with — or even for — a robot.

This article introduces collaborative robotics and discusses several key implementation challenges that readers should consider as they explore this frontier of robotics

Collaboration

A working definition of collaboration incorporates the concept of two or more entities working together to achieve a shared goal more efficiently and/or effectively than would be possible by working independently. A common feature of collaboration includes either explicit or implied rank or status and a corresponding deference to the collaborator in authority. Collaboration is often fostered if collaborators can predict or anticipate the needs of other collaborators, either by task assignment or through experiential learning.

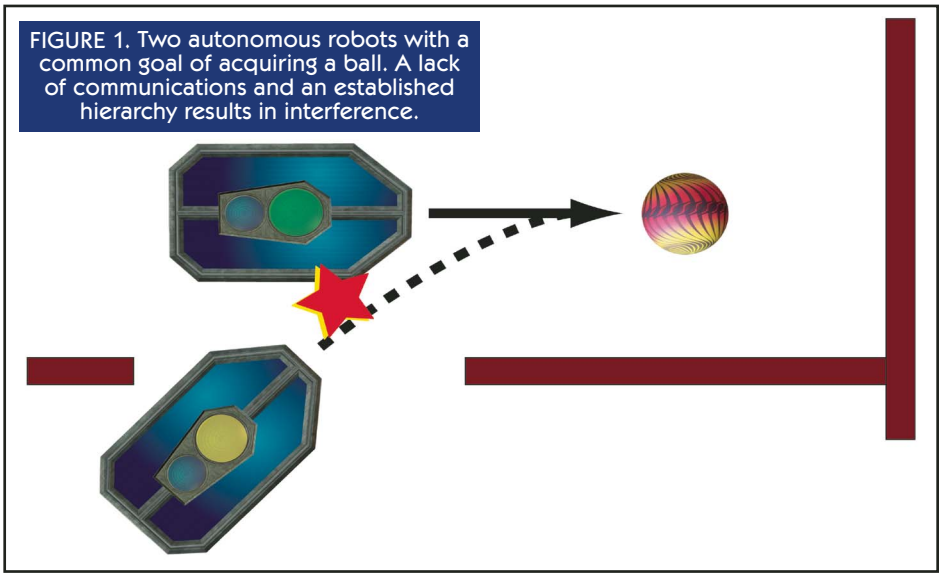
Collaboration isn't necessarily a good thing or even desirable in every situation — there are instances where a single, autonomous individual of action is preferable to a committee of indecisive decision makers. However, there are numerous examples of situations in which efficiency and effectiveness gains can be realized through collaboration, such as hunting in a pack, team sports, warfare, and firefighting.

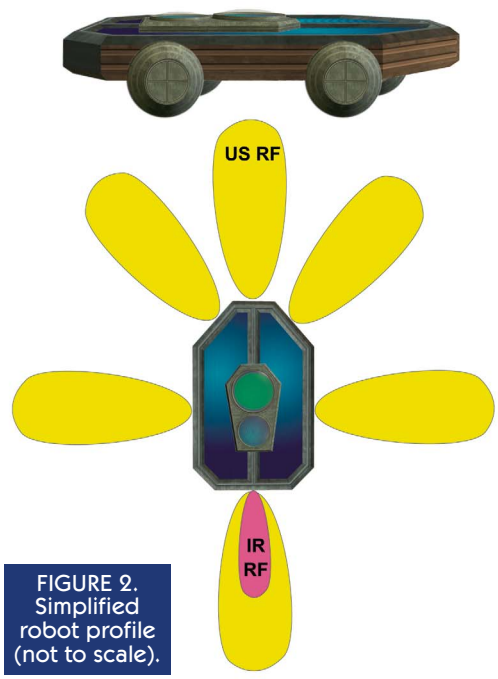
The most publicized example of robot-robot collaboration is the annual RoboCup competition, which is an international proving ground for collaborative robotics and related AI topics [1]. The goal of the organization is to develop a team of fully autonomous humanoid robots that can win against the human world soccer championship team by 2050.

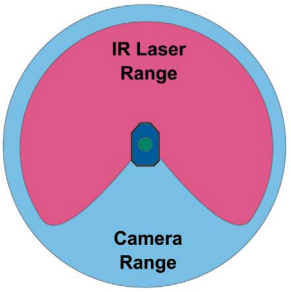
Developments in the more challenging frontier of human-robot collaboration are represented by NASA's Robonaut [2] and, to a lesser extent, DARPA's BigDog [3]. Robonaut is NASA's attempt to create a robotic astronaut capable of working side-by-side with astronauts in the International Space Station. DARPA's BigDog — a robotic replacement for pack animals that is capable of carrying a 100-pound payload over rugged terrain — shares characteristics of both traditional 'slave' robots and collaborative robots.

Collaborative robots typically rely on multiple, powerful computer

FIGURE 1. Two autonomous robots with a common goal of acquiring a ball. A lack of communications and an established hierarchy results in interference.







Dimensions	40 x 30 x 15 cm
Mass	12 kg
Top Speed	1.6 m/s
Endurance	20 minutes
Turning Radius	28 cm
IR Laser Range	3 m
IR Laser Coverage	240°
Camera Range	4 m
Camera Coverage	360°
US RF Range	3 m
IR RF Range	1.5 m
Weapon Range	2 m
Weapon Rep Rate	1/sec
Weapon Kill Radius	1 m

FIGURE 2. Simplified robot profile (not to scale).

processors and algorithms. Even so, biological organisms suggest that heavy iron isn't a prerequisite for robot collaboration. Ants, bees, and many other insects rely on distributed or swarm intelligence to accomplish collaborative feats that researchers have only begun to approximate [4].

Robot-Robot Collaboration

The simplest form of collaborative robotics is collaboration among two robots. However, achieving a true collaborative relationship is far from simple. To appreciate the challenge of designing a pair of robots capable of collaborating on a straightforward task, consider the interaction depicted in Figure 1. Each robot is equipped with a typical suite of autonomous features, including the ability to avoid obstacles, locate objects by color or shape, manage energy stores, and navigate. Furthermore, each robot is programmed to roam and acquire balls that come into sensor range. That is, they have an identical — but not shared — goal.

Returning to Figure 1, assume the robot with the green dome is first to detect the ball. It establishes a direct path to the target and, as it approaches, it monitors the robot-wall distance and robot-ball distance with ultrasonic and IR sensors. The robot with the yellow dome detects the ball at about the same time, but it has a more arduous task to get into position. It moves through the doorway and heads toward the target. However, neither rapidly-moving robot is adequately equipped with sensors or programming to sense the other robot in time. The result is a collision that may damage both robots.

How could the collision have been avoided? More importantly, how can the robots work collaboratively to acquire balls as effectively as possible? One solution is communications. Given adequate robot-robot communications, the first robot to locate a ball can signal the other robot that it is preparing to capture the ball. The second robot would then be free to search for other balls and avoid interfering with the first robot. This very loose collaboration consists primarily of robots staying out of each other's way.

Another solution involves enhancing the sensors so that robots can detect each other from a significant distance and establishing one robot as the 'alpha' robot. Given potential competition over a target, the second robot defers to the alpha robot.

Collaboration can also be enabled by assigning different roles to each robot. For example, one robot is assigned the role of scout, and the other the role of retriever. Role assignment often has an associated cost benefit. In this case, the scout can carry the heavy, power-hungry, and costly sensors, while the retriever can be outfitted with fewer and less costly sensors.

Role assignment, while seemingly straightforward, has a number of dependencies. The first is task decomposition which, in turn, defines necessary robot competencies. A robot tasked with scouting for possible targets may spend most of its time roaming, following walls, entering doorways, and perhaps periodically notifying the other robot of its position. The most important scout robot competencies are related to spotting targets from a significant distance, mapping the area, and communicating the target coordinates — and perhaps even the most efficient route — to the retriever.

In contrast, the retriever might spend most of its time in an energy-saving mode, awaiting a signal from the scout before activating its drive system and navigation sensors. Its primary competencies involve receiving and interpreting communications from the scout, following the suggested path to the target from its current location, and efficiently acquiring the target.

The competencies of both robots must be supported by the appropriate sensors, effectors, hardware platform, and operating system. Borrowing a technique from game development, it's helpful to create a robot profile for each robot, as in Figure 2. The profile should contain graphical and tabular data on sensor range and function and basic physical capabilities of each robot. In this simplified profile, the robot is a modest wheeled robot, relatively low to the ground, and equipped with an omni-camera, IR laser rangefinder, as well as discrete US and IR sensors.

Tight Collaboration

In the previous example of a loose collaboration, robots needn't ever come in close proximity to each other, except by accident. In contrast, with tight collaboration, robots work interactively to achieve a shared goal. Consider the scenario depicted in Figure 3, in which a pair of robots collaborate in hunting down a human player in a game of laser tag. The shared goal is to pursue, trap, and neutralize the human shooter while avoiding being tagged.

As demonstrated by say, the pursuit of a deer by a pair of wolves, collaborative pursuit is a complex, carefully choreographed interaction in which the smallest mistake on the part of the predator can shift the advantage to the prey. Whereas a single predator may approach its prey from behind and at a slight angle to encourage the prey to follow an inefficient arc in an attempt to escape, two predators have many more options. One predator can wait in ambush, while the other predator maneuvers the prey into position. Alternatively, one predator can distract the prey while the other predator takes up a superior position. The two predators could opt to simply surprise and confuse the prey by appearing at the same instant.

In each case, the coordinated behaviors require planning, communications, and the ability to predict not only how the prey will respond to a threat, but how the collaborating robot will respond to changes in prey behavior. Prediction is an important capability, and one not easily satisfied without an internal model of how collaborating robots will react to potentially novel situations. Predicting human prey behavior is particularly challenging.

In a tight collaboration scenario, the robot profiles are crucial for understanding the interplay of the robot sensors with the environment, the target, and the sensor signals from collaborating robots. For example, the two robots approaching the entrance of a room holding the hostile in Figure 3 are painting each other with ultrasound and IR signals. Even though the ranging distance might be limited to a meter or two, the transmitted IR and US signals travel hundreds of meters and reflect off of walls and other structures. The result is an increase in the noise floor, which appears as random variation in distance measures. Furthermore, if the ultrasound sensors are receptive to reflected signals, a sensor on one robot can be triggered by the direct or reflected signal from another transducer. As the number of robots increases, so does the noise and risk of sensor saturation and false triggering.

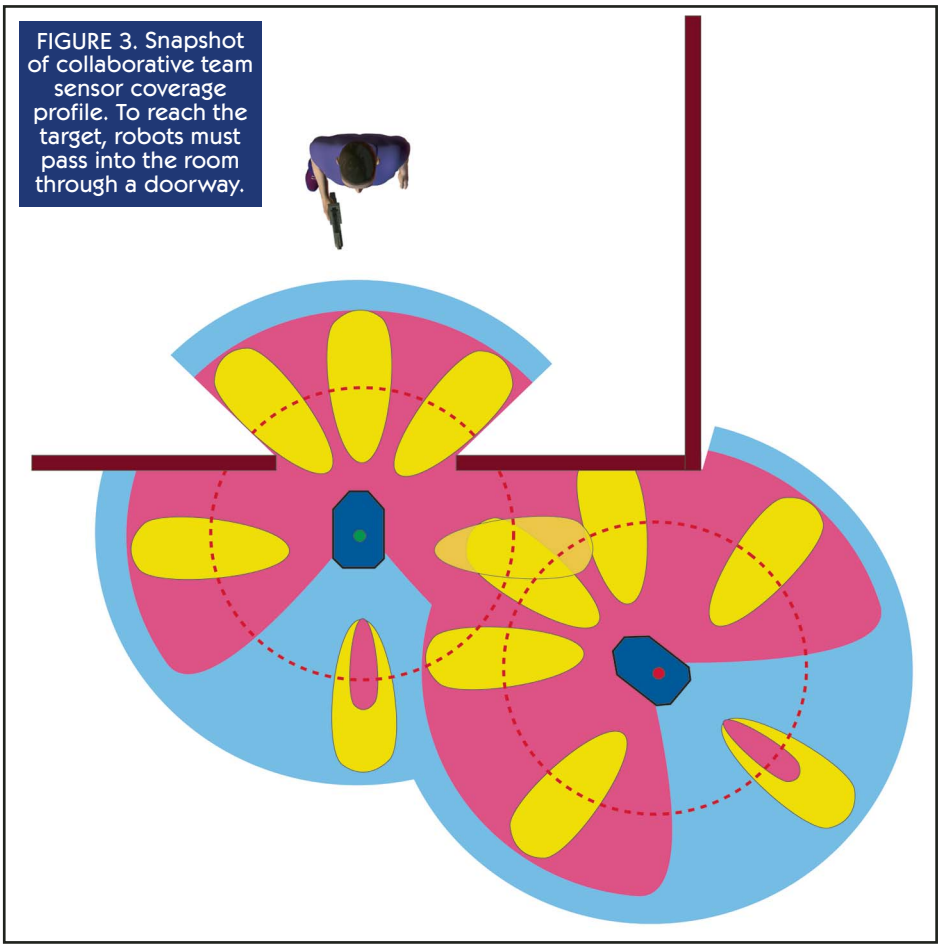


FIGURE 3. Snapshot of collaborative team sensor coverage profile. To reach the target, robots must pass into the room through a doorway.

Human-Robot Collaboration

Human-robot collaboration — for years limited to the realm of science fiction — is the real frontier of robotics. In addition to the capabilities required for robot-robot collaboration, the human

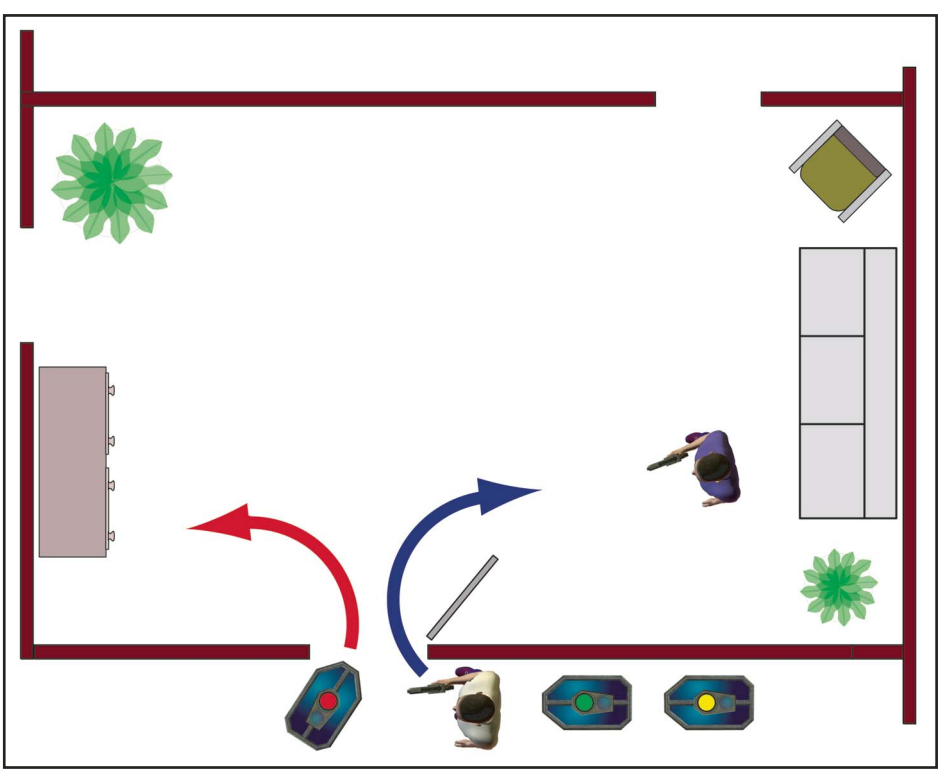


FIGURE 4. Human-robot collaboration in a game of laser tag involving coordinated entry into a room with a hostile (blue shirt).

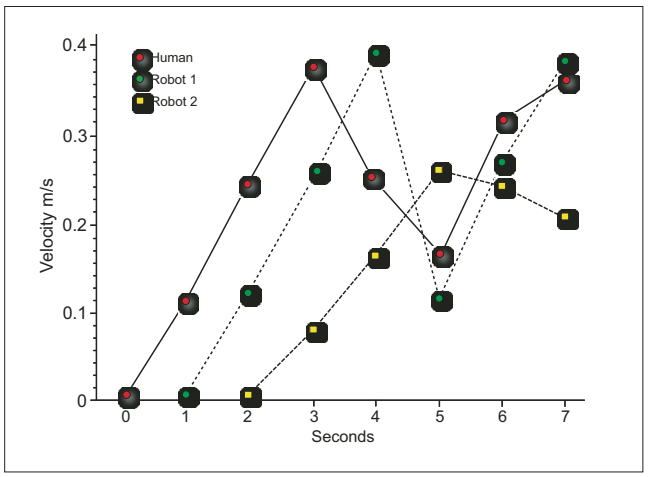


FIGURE 5. Velocity plot associated with a collaborative human-robot team walking in formation toward a target.

element demands new ways of thinking about robots and of human relationships.

Figure 4 depicts a scenario in which a team of three robots and one human (white shirt) are playing a game of laser tag against another human (blue shirt). At first glance, neutralizing the outnumbered hostile is simply a matter of entering the room, identifying the hostile, and firing.

However, on closer examination, not only is entering the room without creating a pileup in the doorway a highly complex maneuver, but simply assembling in an orderly fashion outside the door is beyond the capabilities of most autonomous robots. As in two-robot collaboration, role assignment, task decomposition, and modeling collaborator and prey responses are critical.

Simply adding a third robot to the team increases complexity exponentially. For example, consider what's involved in expanding intra-team communications from two to three robots. With two collaborating robots, there is no need for a robot to identify itself or the intended recipient of a message. With three robots, there is message ambiguity without a means of identifying the source of each message because messages can be generated by one of two other robots. Furthermore, communications can be broadcast to the other two robots or, given the proper communications equipment, a robot can be singled out to receive a private message. Robots must also contend with the noise and interference from three robots in close proximity generating audio, RF, US, and IR noise.

Let's explore the challenge of walking in formation toward the room entrance. An autonomous robot simply contends with following the wall, identifying the entrance, and turning into the doorway as soon as the edge of the entrance is recognized. Moving in formation requires a new set of behaviors and competencies related to leading, following, robot-robot and robot-human spacing, and velocity modulation.

As shown in Figure 5, one scenario is for the team leader — in this case, a human — to initiate the approach. The first robot (green dome) reacts to this movement by accelerating toward the leader, and the second robot (red dome) similarly reacts to the movement of the first robot. However, depending on the acceleration characteristics of the human and robots, response time of sensors, and target spacing, the formation can move in unison, like an oscillating slinky, or the first robot could ram into the ankle of the lead human and the second robot follow up by crashing into the first robot.

The velocity plot shown in Figure 5 illustrates a responsive first robot that tracks human velocity with a lag of about one second. It both over-and under-shoots the human's velocity. The second robot responds poorly to the first robot's movements, and has difficulty maintaining a constant robot-robot distance. The solution might involve equipping the second robot with a more capable quad encoder and PID configuration and perhaps replacing brushed motors with a more predictable brushless design.

Designing robots capable of supporting human-robot collaboration is especially challenging when the team is intended to interact with humans. A human target in the laser tag game introduces the concept of friend versus foe recognition. The first step is to identify the humans in a room. This

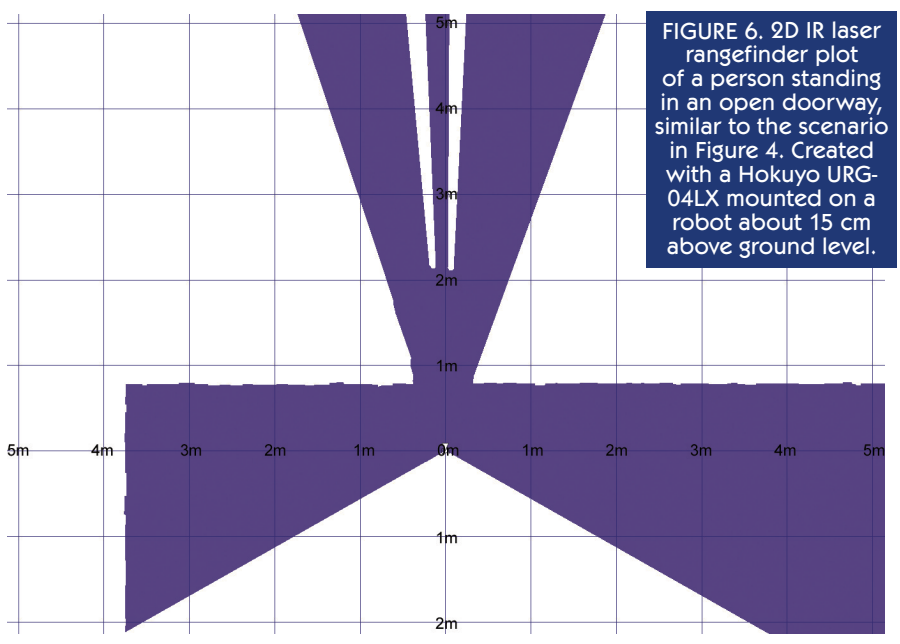


FIGURE 6. 2D IR laser rangefinder plot of a person standing in an open doorway, similar to the scenario in Figure 4. Created with a Hokuyo URG-04LX mounted on a robot about 15 cm above ground level.

may require reinterpreting, augmenting, or completely replacing sensors. For example, the IR laser used to identify the human target in the robot-robot collaboration example (Figure 3) is inadequate for differentiating between the human team member and the human hostile.

As shown in Figure 6, a human friend or foe facing a relatively short robot equipped with an IR laser rangefinder appears as two shadows — one for each leg. It's even more challenging to identify a human standing perpendicular to the robot, since the legs produce only a single shadow. In either case, differentiating friend from foe based on leg shadows alone isn't possible. One solution is to configure the omnidirectional camera to recognize colors, assuming the hostile and team member consistently wear blue and white shirts. Fusing the IR laser rangefinder and color data could provide a more accurate indication of friend and foe position.

Another challenging task is tracking the lead human in the initial lineup prior to entering the room with the hostile. The laser rangefinder used to create Figure 6 provides a 240 degree field-of-view, updated at 10 Hz. While this refresh rate may be sufficient for mapping purposes, it is inadequate for tracking the rapidly moving feet of the lead human, especially if the human stops or turns unexpectedly.

There is also the issue of team communications. Candidates for human-robot communications range from a simple, multi-button RF signaling device to more computationally intensive gesture or voice recognition.

Performance constraints are also critical in human-robot collaboration. Although the relationship will eventually evolve, current performance standards are imposed by humans. A human can travel and respond faster to visual cues than any commercial robot. In an operation such as depicted in Figure 4, the robots limit the pace of the operation.

From Here

Collaborative robotics is beginning to appear in military and even some commercial offerings. For example, robotic surgical assistants now routinely assist surgeons performing prostate surgery in hospitals throughout the US. The caveat is that the relationship is the typical master-slave relationship of old — which makes sense, given that surgeons aren't likely to invest in technology intended to replace them — a fear highlighted by Ellison's classic short story *Wanted in Surgery* [5].

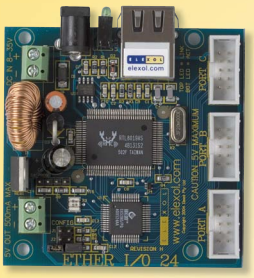
This fear of robots empowered with collaborative leadership qualities is being addressed head-on by research in the area of sociable robotics. Although limited to research experiments, sociable robots have demonstrated the value of robots that can understand and relate to humans in a personal way.

One of the best known sociable robots — MIT's Kismet [6] — illustrates the limitation of current technology. Kismet is more animatronic than robotic, in part because it relies on a network of 15 PCs to enable it to exhibit a modicum of social intelligence. In time, multi-core parallel processing and new algorithms will bring socially intelligent, collaborative behaviors to every robot. For now, collaborative robotics is a practical goal that is well within the reach of every enthusiast willing to take up the challenge. SV

www.orteches.com www.orteches.com www.orteches.com www.orteches.com www.orteches.com www.orteches.com

NEW! ELEXOL Ether I/O 24

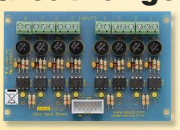
UDP/IP controlled digital input/output module featuring three 8-bit ports with 5V level signal lines. Each of the 24 lines can be independently programmed as either an input or an output. Connects to any TCP/IP protocol network.



- Supports ARP, BOOTP, DHCP, ICMP and UDP/IP protocols
- Standard 10BaseT Ethernet Interface; RJ45 connector
- 24 independently programmable signal lines with configurable CMOS, TTL or Schmitt Trigger thresholds and programmable pull-ups per line
- Easy connection by three 10-way box headers
- On-board 50MIPS flash micro-controller may be reprogrammed to suit specific applications
- Integrated switch-mode voltage regulator allows power supplies from 8-32VDC
- User 5V 500mA output to power external interface boards or sensors
- Compact module – 72 x 72 x 24mm

Great range of I/O-24 peripherals, too...

CONNECTOR/LED BOARD
Provides screw terminal connections plus optional either pull up or pull down resistors and LED port status indication.



OPTO INPUT BOARD
Provides 8 Opto isolated inputs for each of the I/O-24 pins with LED indications.

RELAY OUPUT BOARD
Provides 8 isolated relay contact outputs suitable for a variety of loads.

SWITCH/PUSH BUTTON BOARD
Provides 8 switch/push buttons with LED indicators showing the status of the I/O pin.

Visit our website www.orteches.com

Ortech™ Education Systems

Ortech Education Systems
1119 - 9th Ave. S. • Moorhead, MN 56560
218.287.1379

www.orteches.com www.orteches.com www.orteches.com www.orteches.com www.orteches.com www.orteches.com

REFERENCES

[1] Nardi, D., M. Riedmiller, et al., eds. RobCup 2004: Robot Soccer World Cup VIII. 2005, Springer: New York.

[2] Ambrose, R. Robonaut. 2006 [cited 2006 June 13]; Available from: **vesuvius.jsc.nasa.gov/er_er/html/robonaut/robonaut.html**.

[3] Hambling, D. Robotic 'pack mule' displays stunning reflexes. 2006 [cited 2006 May 15]; Available from: **www.newscientist.com/article.ns?id=dn8802**.

[4] Tarasewich, P. and P. McMullen, *Swarm intelligence*. Communications of the ACM, 2002. 45(8): p. 62-7.

[5] Ellison, H., *Wanted in Surgery*, in If. 1957, Quinn Publishing Company: Buffalo, NY. Audio version available on **Audible.com**.

[6] Breazeal, C., *Designing Sociable Robots*. 2002: MIT Press.

COMBOTS CUP

Win $10,000!

Combat Robots are back and better than ever! Charge your batteries and register for the second annual ComBots Cup. The middleweight champ will get $3,000 and the heavyweight champion will take home $10,000! And of course, the ComBots Cup itself - 100 pounds of metallic glory.

Last year, over twenty robots fought to the death for the cup, with number one ranked Sewer Snake finally winning. This year the Cup and money could be yours. You've got three months. Start building!

January 14-15, 2007 in Oakland, CA
http://ComBots.net or cup@combots.net

ROBOT RFL FIGHTING LEAGUE

SERVO Magazine on CD!

2005 CD-Rom Now Available!

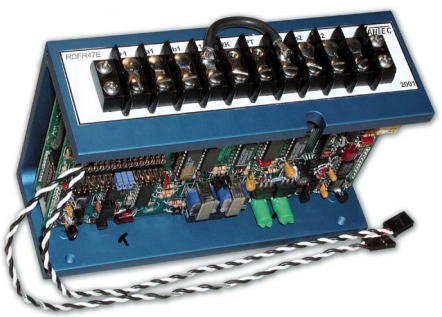
Great news! Now you can own the 2nd published year of *SERVO* on CD. Stored as PDFs, you can print, search, and easily store your copies on digital media.

You can purchase said CD from us, for your own personal use, and donate your old paper copies to your local school or library. (Your spouse will love you for it!)

This makes a great gift, too! Give it to a friend or loved one to show how much you care!

To order, call: 1.800.783.4624

STEER WINNING ROBOTS WITHOUT SERVOS!



Perform proportional speed, direction, and steering with only two Radio/Control channels for vehicles using two separate brush-type electric motors mounted right and left with our **mixing RDFR dual speed control**. Used in many successful competitive robots. Single joystick operation: up goes straight ahead, down is reverse. Pure right or left twirls vehicle as motors turn opposite directions. In between stick positions completely proportional. Plugs in like a servo to your Futaba, JR, Hitec, or similar radio. Compatible with gyro steering stabilization. Various volt and amp sizes available. The RDFR47E 55V 75A per motor unit pictured above. **www.vantec.com**

VANTEC

Order at (888) 929-5055

Extreme Robot Speed Control!

Sidewinder

$399

- 14V - 50V
- Dual **80A** H-bridges 150A+ *Peak!*
- Adjustable current limiting
- Adjustable speed slew rate
- Temperature limiting
- Three R/C inputs - serial option
- Many mixing options
- Flipped Bot Input
- Rugged extruded Aluminum case
- 4.25" x 3.23" x 1.1" - Only 365g

$29.99

Scorpion Mini

- **2.5A** (6A pk) H-bridge
- 5V - 18V
- 1.25" x 0.5" x 0.25"
- Control like a servo
- Optional screw term.
- Only 5.5g

$79.99

Scorpion HX

- Dual **2.5A** (6A pk) H-bridges
- Plus **12A** fwd-only channel
- 5V - 18V
- 1.6" x 1.6" x 0.5"
- Four R/C inputs
- Mixing, Flipped Bot Input
- Only 22g

$119.99

Scorpion XL

- Dual **13A** H-bridge *45A Peak!*
- 5V - 24V
- 2.7" x 1.6" x 0.5"
- Three R/C inputs - serial option
- Mixing, Flipped Bot Input
- Only 28g

$199

OSMC - Monster Power H-bridge

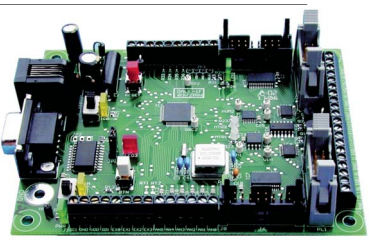
- 14V - 50V and **160A** over **400A** peak!
- 3.15" x 4.5" x 1.5"
- Control with Stamp or other Micro
- 3 wire interface
- R/C interface available

All Robot Power electronic products are proudly MADE IN THE USA

Introducing Dalf

Advanced dual motor drive with closed-loop control functions

Only $250



Embedded Electronics, LLC along with our exclusive reseller Robot Power are proud to introduce a feature rich, customizable Dual Motor Controller: *Dalf*. Designed to work out of the box or to host your application specific code; Dalf makes it simple to create a complete turn-key "brain" for your application with full-closed-loop motion control. Just take a look at these features!

Motion Control Functions

Drives all sign-magnitude brushed DC motor drives such as the OSMC

Closed-Loop Features

- Closed-loop control of two motors
- Full PID position loop
- Trapezoidal path generator
- Adjustable slew rate for smooth transitions
- Non-volatile storage of PID parameters
- Step-Response PID motor tuning support
- Quadrature encoder support for each motor
- Terminal mode for interactive tuning and debugging
- Windows GUI under development

Open-Loop Features

- Two R/C command modes (3 input channels)
- Two open-loop pot control modes
- Interactive terminal control of motors
- Adjustable slew rate

Application Support

- PIC18F6722 CPU running at 40MHz
- 60k+ FLASH available
- Serial bootloader, no programmer needed
- Serial command/monitor in both terminal and high-speed binary API mode
- I2C slave command interface
- Firmware implented in C andASM
- C source for main loop and utility routines provided free
- Linkable device driver function library provided for building custom applications
- Extensive documentation with Owner's Manual and Getting Started Manual provided on CD
- Custom code development services available (contact EE)

I/O Connections

- Two RS-232 serial ports
- 36 GPIO
- I2C master and slave ports (2 ports)
- Two motor drive outputs
- Two quadrature encoder inputs
- Two Hall-effect current sensors inputs
- Six 10-bit A/D
- Two channels of cooling fan control
- Standard ICD connector

For more *Dalf* information visit www.embeddedelectronics.net

ROBOT POWER™

www.robotpower.com

Phone: 253-843-2504 • sales@robotpower.com

Also from Robot Power

Kits, parts, schematics
Planetary gearmotors

DARwIn

PART 1: Concept and General Overview

by: Karl Muecke, Patrick Cox, and Dennis Hong
RoMeLa (Robotics & Mechanisms Lab) at Virginia Tech;
www.me.vt.edu/romela

DARwIn (Dynamic Anthropomorphic Robot with Intelligence) is a humanoid robot capable of bipedal walking and performing human-like motions. Developed at the Robotics & Mechanisms Laboratory (RoMeLa) at Virginia Tech, DARwIn is a research platform for studying robot locomotion and also the base platform for Virginia Tech's first entry to the 2006 Robocup competition (Figure 1). First started as a design project for seniors in the department of mechanical engineering at Virginia Tech, the general goal in designing DARwIn was to create a bipedal robot with human proportions; suitable to be a research platform capable of dynamic walking. The 600 mm tall, 4 Kg robot has 21 degrees-of-freedom (DOF) with each joint actuated by coreless DC motors via distributed control with controllable compliance. Using a computer vision system on the head, rate gyros in the torso, and multiple force sensors on the foot, DARwIn will be able to implement human-like dynamic gaits while navigating obstacles and traverse uneven terrain while implementing complex behaviors such as playing soccer.

This three-part series will describe the design and fabrication process for the first version of DARwIn's hardware, highlight the salient features,

< FIGURE 1. Robocup 2006 participants (DARwIn not pictured). Photo courtesy of Messe Breme **www.robocup2006.org**.

and briefly discuss some of the research issues related to creating the walking gait and control algorithm for the robot.

Robocup 2006

For those of you unfamiliar with Robocup, it is an international soccer competition between many classes of robots, including a four-legged league (Figure 2) and a humanoid league (Figure 1). By 2050, the organizers of Robocup hope to have fostered a team of humanoid robots that can defeat the human World Cup champions in soccer (**www.robocup.org**)!

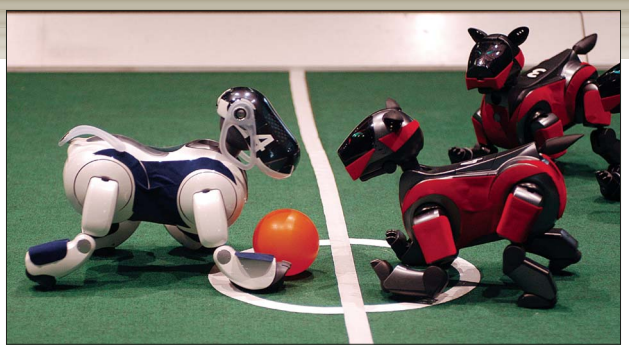
The challenges in the last humanoid league Robocup were: a two-on-two soccer game, a penalty shootout, uneven terrain navigation, passing, and dribbling. There are two divisions within the humanoid league: teen size and kid size. The size is defined as the smallest number between the robot's height to the top of its head and 2.2 times the center of mass height:

$$H = \min (H_{Top}, 2.2 \times H_{COM})$$

For 30 cm ≤ H ≤ 60 cm, the robot is kid size and for 65 cm ≤ H < 180 cm, the robot is teen size. We plan to have DARwIn compete in the kid size competition of the humanoid league of Robocup and be the first US team to even compete in the humanoid division of Robocup.

Imitating a Ball and Socket Joint

Since the design and construction of humanoid robots is still in its infancy, a robot with superior structural and kinematic design, even with moderate AI, would prove to be a very formidable opponent in Robocup. With this in mind, we designed a feature rarely seen in humanoid robots — kinematically spherical joints. A human's shoulder and hip (among other joints) are ball and socket joints, which allows for three degrees of freedom in rotation about a single point. Servo



> FIGURE 2. Four-legged division of Robocup 2006. Photo courtesy of Messe Breme **www.robocup2006.org**.

motors are great for their price, power, and size, but are only capable of one degree-of-freedom axial rotation. Even with this limitation, three motors arranged so that all three axes always intersect at a single point are kinematically equivalent to a true spherical joint (Figure 3). In this configuration, the robot's joints are kinematically identical to a human's ball and socket joint.

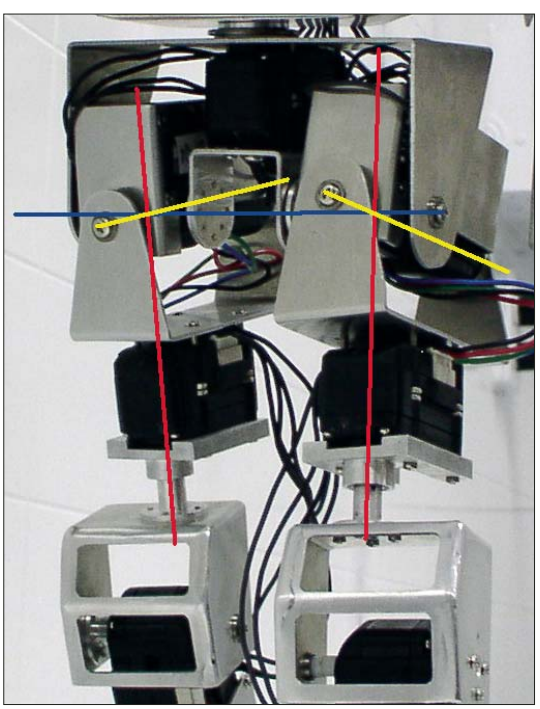
Why have a spherical joint? If all of the axes did not intersect at the same point, the offsets would cause the link to change its effective length as the joints move. For example, in the case of the hip joint, certain rotations of joints could cause the overall length of the leg to increase or decrease. In "general robotics," this is normally handled easily by the forward and inverse kinematic equations.

However, for a dynamic walking robot where the inertial forces of the moving links have a great effect on the overall motion and stability of the robot, having kinematically spherical joints not only allows for simpler kinematic modeling, but also significantly simplifies the dynamics and control, which is essential for the onboard computer to process in real time.

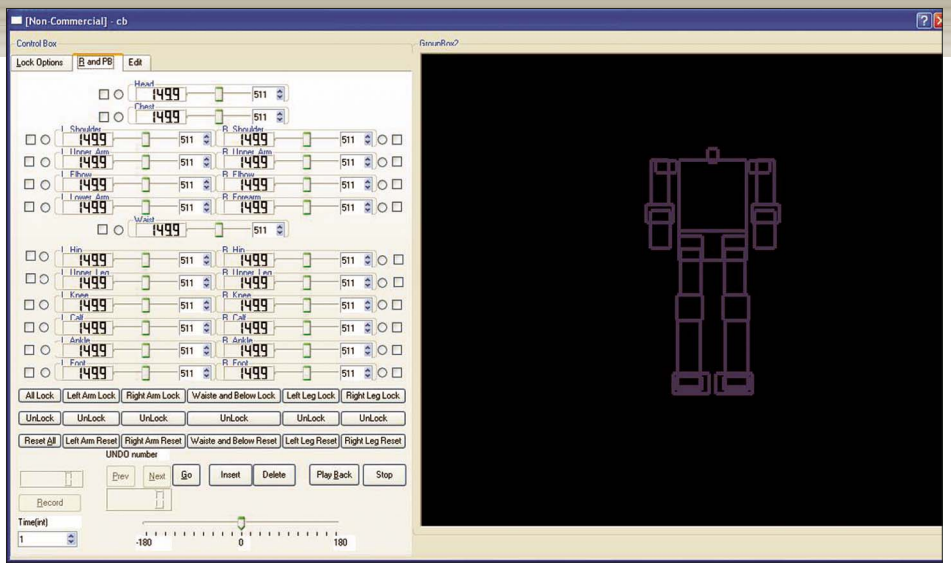
Dynamic vs. Static Walking

In addition to imitating the structural design of humans, we are also looking into how humans walk. Humans walk in a dynamic fashion, a state of constant falling, where our center of mass (COM) is not always over our foot. When we pick up our foot and walk forward, we actually fall forward and catch ourselves as we step.

Unlike humans, most bi-pedal robots walk using the static stability criterion, which always keeps the COM over the support polygon — the area formed by connecting all of the outer most points of the robot (usually the foot/feet) in contact with the ground. The robot stands on one foot and moves the other while keeping the COM over the grounded foot by constantly adjusting its posture. When both feet are on the ground, the robot can shift its COM over the forward foot. The robot repeats this process for each step. We hope to have DARwIn walk as humans do — dynamically. Dynamic walking is generally a faster and much more efficient mode of walking. However, implementing a dynamic gait with a bipedal robot is no easy task.



> FIGURE 3. Close view of hips with lines through axes of rotation to show that they all intersect at the same point for both hips.



< FIGURE 4. Screen shots of the in-house developed code for LabVIEW (top) and C++ software.

CM-2 Board, C++, and LabVIEW

The method of controlling DARwIn's walking and sensing has gone through a few iterations. Initially, DARwIn used Robotis Co.'s CM-2 controller board to control the DX-117 motors, from the same company. The DX-117 motors are special servo motors that use Maxon's efficient Re-Max DC motors and a RS-485 serial network for distributed control. We will describe more about the use of these motors in our next article.

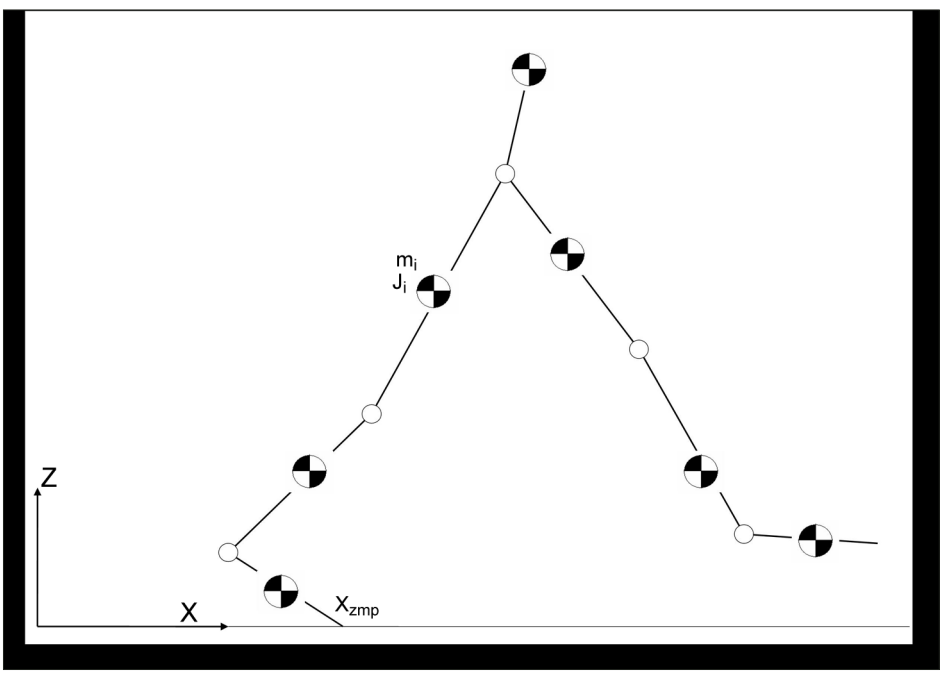
The CM-2 board, which uses an Atmel ATMega 128 microcontroller, was great for prerecording the robot's motions and playing them back to generate the walking gait, but seriously lacked computing power and memory space to implement real-time control. To enable DARwIn to walk dynamically, the computation requirements became higher for processing all the input from various sensors and implementing complex algorithms. Thus, we developed our programs using both C++ and LabVIEW (Figure 3) running on a laptop tethered to DARwIn.

In the next version of DARwIn, an onboard PC/104 single board computer will replace the laptop with the tether. Currently, each platform performs successful walking using record and playback motion and has an OpenGL model of the robot, which is a 3-D representation of the robot on the laptop screen shown in real time. Using LabVIEW for the software makes it very easy to program and implement various control strategies due to its easy interface and graphical programming approach. However, LabVIEW's overhead for the computer is high.

On the other hand, using C++ code for the software makes it much more efficient, but the lengthy development time and steep learning curve could be problematic for beginners. Both programs use the syntax provided in the motor's documentation to poll all of the motor's positions. The positions are stored and based on a speed or time step; the velocity for each motor can be calculated as

$$V(k) = (\text{Position}(k+1) - \text{Position}(k))/T_s$$

Using the velocities calculated and the positions stored, the program sends all of the motors position and velocity information so that all of the motors move to position(k+1) at velocity(k), and then repeats the process after time step T_s. Our current version of the software does not implement acceleration control.



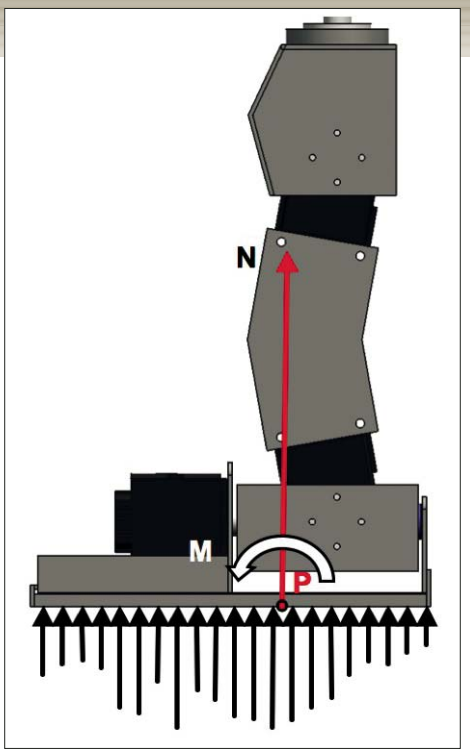
< FIGURE 5. Diagram of a bipedal robot.

Zero Moment Point

DARwIn needs a more powerful hardware platform than the CM-2 board because it needs to solve many equations in order to determine the location of the zero moment point (ZMP). The ZMP is where all of the moments in a body sum to zero. It can be calculated by

$$x_{ZMP} = \frac{\sum_i m_i(-\ddot{z}_i x_i + (\ddot{x}_i + g)x_i) - J_{iy}\ddot{\theta}_{iy}}{\sum_i m_i(\ddot{z}_i + g)}$$

where m_i is the mass of each link of



> **FIGURE 6.** Diagram showing the ZMP at reaction point P.

the robot; x_i is the x position of the COM of each link; z_i is the z position of the COM of each link; g is gravity; J_{iy} is the moment of inertia about the y axis of the COM of each link; and θ_i is the angular position of each link in Figure 5. When at least one leg is in contact with the ground, the ZMP exists at the Center of Pressure (COP) seen in Figure 6. By keeping the ZMP within the support polygon of the robot, we can maintain stability with the COM outside of the support polygon and allow for dynamic walking.

Moment of Inertia

One of the trickiest parts of developing a bipedal robot that can walk dynamically is determining the moment of inertia (MOI) of each link of the robot. One of the terms in the ZMP equation is J, or the MOI of the link. This value is needed to calculate the ZMP, which is critical in achieving stable dynamic walking of the robot. To determine the value of MOI, we built a device that allows us to calculate the MOI of each link (Figure 7).

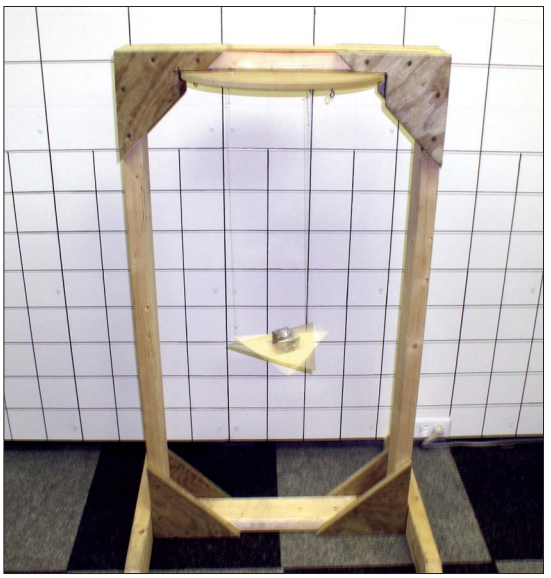
The device consists of a sturdy stand, three equal lengths of string/cord, and a strong lightweight platform to place the part on. By placing the part on the platform, the period of oscillation of the device changes according to what the MOI of the part is. So, by measuring the new period of oscillation, we can determine the MOI of each part.

Next Month ...

You've seen DARwIn's kinematically spherical joints, higher-level control approaches, and some theory for dynamic walking that will be used. Next month, you will see how we built DARwIn, including details on motors, wiring, parts, force sensors, and the fabrication process. Read next month's article to find out how you can build your own bipedal robot. **SV**

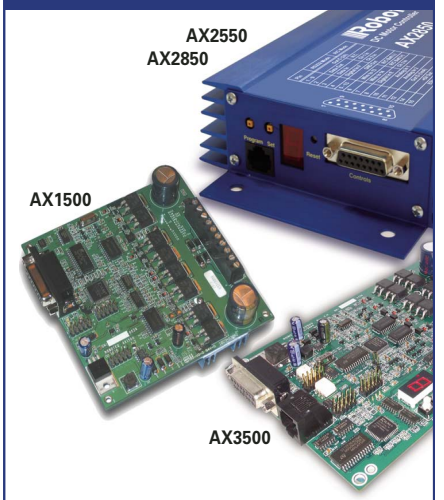
Thank Yous

Thanks to the 2005-2006 senior design team members that built DARwIn 1.0 (Patrick Cox, Joo Gil, Chris Greenway, Jeff Kanetzky, Karl Muecke, Patrick Mulliken, Raghav Sampath, and Daniel Zokaites) and to our advisor, Prof. Dennis Hong.



> **FIGURE 7.** Picture of test setup to measure moment of inertia.

Industrial Strength Motor Control for All



Get the DC Motor Controllers that are at the heart of many of today's world's most demanding Industrial, Military and Research Robots, and other innovative Motion Control applications.

- RS232, RC, or Analog input
- Speed or Position Mode
- Dual channel output up to 140A
- Optical Encoder Inputs
- Intelligent Current Limiting
- Thermally Protected
- Field Upgradable Firmware
- Enclosed and Board-Level versions
- and many more advanced features ...

Model	Amps	Features	Price
AX1500	2x30A	B	$275
AX3500	2x60A	O-R-B	$395
AX2550	2x120A	A	$495
AX2550HE	2x140A	A	$645
AX2850	2x120A	O-A	$620
AX2850HE	2x140A	O-A	$770

A=Aluminum Extrusion, B=Board-Level, O=Optical Encoder In, R= RC outputs. Qty1 price. Contact us for OEM Qty prices

RoboteQ

8180 E.Del Plomo Dr.
Scottsdale AZ USA 85258
(602) 617-3931 - info@roboteq.com

www.roboteq.com

Interface Your PC to an R/C Radio

by Michael Simpson

Recently, I built a couple of projects where I interfaced a PlayStation controller to a microcontroller. While this works fine in many situations, there are times when you want to use an R/C radio for its extended range or reliability. In this article, we are going to create a very accurate and reliable interface to any R/C radio with up to six channels. To perform this interface, we will create a coprocessor using a microcontroller. Once complete, we can then connect other devices or even a PC to this interface.

THE RC RADIO

An R/C receiver has multiple connectors that allow you to connect to servos or speed controllers. Each of these connectors has three leads. Two of the leads provide power and the third is a signal lead. The signal lead provides a positive pulse once every 20 milliseconds. It is the width of this pulse that each servo or speed controller uses to determine its position or speed. On average, the pulse will range from 1,000 to 2,000 microseconds. The neutral or center position is at, or close to, 1,500 microseconds.

As you change the positions of the joysticks or knobs on the transmitter, the pulse width will change in proportion to the amount of movement. When it's all said and done, it is these pulse widths that we are going to measure with our interface.

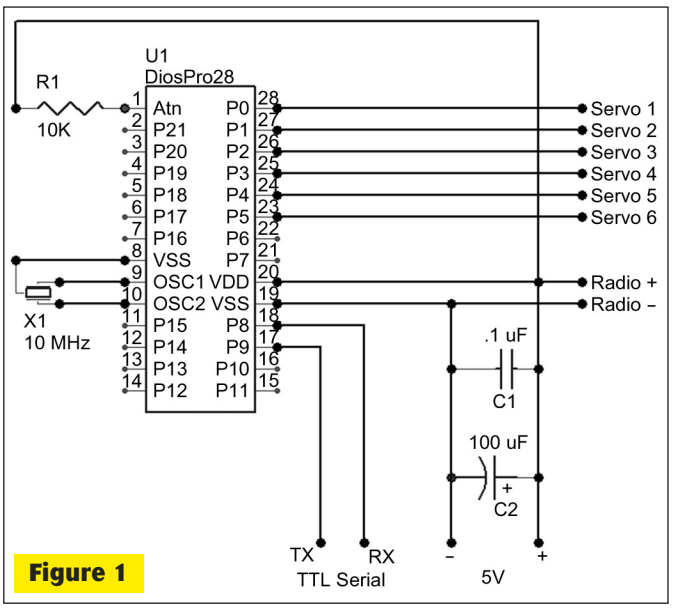


Figure 1

THE OBJECTIVE

When I started the interface project, I came up with a list of objectives:

1) Must support one to six channels.
2) Must cover the pulse range from 1,000 µS to 2, 000 µS at a resolution of 10 µS.
3) The interface must support the PC and microcontrollers.

Since the pulse repeats once every 20 milliseconds, you will never be able to measure the pulses any faster than that. If we take the time to measure each pulse independently one after another, it would take as long as 120 milliseconds to measure six pulses. This is much too slow. What we have to do is measure all the pulses at once. We need a real fast chip to do this. This is where the DiosPro comes in. In order to obtain the resolution we need for six channels, we need to use assembly language. The DiosPRo chip supports inline assembly language with the KRAssembler.

The DiosPro chip also has a built-in UART so we can add a serial interface that just about any chip, device, or PC can talk to.

Figure 1 shows the minimum chip configuration in order to measure

FOR YOUR INFO

Zeus is a simple Windows programming environment that specializes in interface design. A free version of the software called ZeusLite is available at www.KRMicros.com. The actual compiled applications are also provided for those who don't want to play with the code.

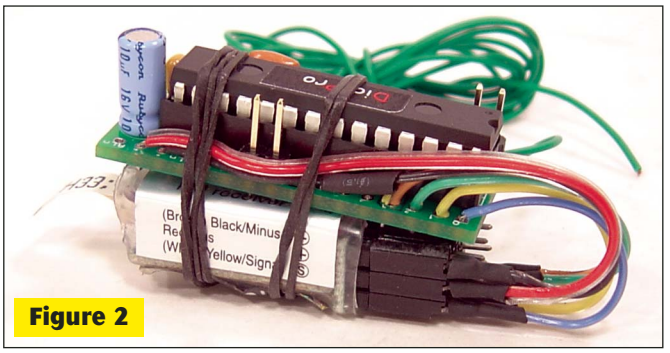


Figure 2

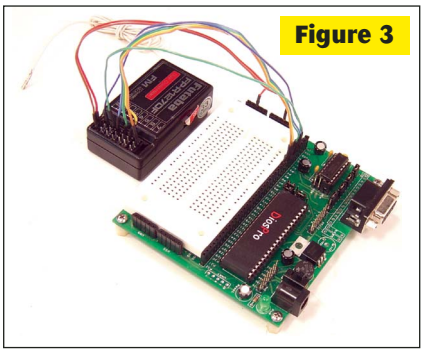


Figure 3

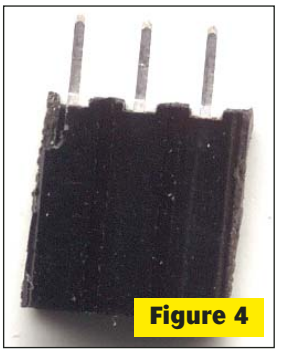


Figure 4

the pulses. To program the chip, you will also need to connect an EZRS232 driver. This driver can also be used to convert the serial TTL levels for connection to a PC.

Take a look at Figure 2. Here, I have piggybacked a small four-channel receiver and the whole thing weighs less than 20 g. While our final goal is to create the smallest payload possible, I recommend starting your interface with a larger board to allow easier prototyping and perfecting of your program.

The Dios Workboard Deluxe shown in Figure 3 is a perfect way to prototype this project. It contains a built-in regulator and RS-232 interface. The built-in interface can be used to program the DiosPro, as well as to provide an interface to the PC. If you have a sizable junk box, you may want to build your own Dios Workboard. For that I have made the PCB available as well. (See Sources sidebar.)

THE FIRMWARE

The program that we program into a microcontroller is often called firmware. The firmware for this project is called RCRadio.txt. It is available — along with the other source and executables — from the Kronos Robotics website (**www.kronosrobotics.com**).

The RCRadio.txt program gives a range of 10 µS to 2,550 µS in 10 µS units. It will return a reading of 150 for a radio pulse of 1,500 µS. The actual accuracy is within the 20 µS we specified earlier, with an accuracy of 10 µS being more typical.

Due to space considerations, the code won't be presented here, but there are plenty of comments if you want to make modifications to the code.

We use a couple of high-level Dios commands to set things up then jump into the KRAssembler with the startasm command.

The bulk of the program takes place within a very tight and controlled loop. During each iteration, we test each of the six ports and if it is high, we increment a counter. Once a channel goes high, we set a corresponding bit in the flag variable called f0.

During the loop, we take a peek at the UART to see if any data has been loaded. If it has, it is quickly removed and loaded into the tcmd variable which we will use later. This small peek is carefully timed so that the same amount of time is spent whether we receive data on the UART or not.

Once the UART is checked, the program then checks the flag variable f0 to see if all the pulses on the indicated channels have been measured. If all the channels have been measured, we check to see if the tcmd variable contains a value of 128. If one was made, we jump to the appropriate routine to transmit the indicated measurements. We then clean up and wait for all the ports to go to a low state before we re-enter the main loop and start measuring again.

Note that even if we are only measuring one port we still cycle through all six. This keeps the numbers and timing consistent.

The baud rate is set to 115200, 8N1. A slower baud rate can be used if you plan on connecting a microcontroller that does not support speeds that high.

To load the firmware into the DiosPro chip, you have two choices. You can use the free Dios compiler software available on the Kronos Robotics website or you can use the ZeusPro software to program the precompiled ZPU file. You will need an RS-232 driver or a Dios Workboard, as well.

CONNECTING THE RC RECEIVER

The connections from the Dios chip to the RC receiver will make or break your project. The following is the most reliable method I have found in doing this.

You will need several three-pin headers — one for each channel. These are created by snapping (cutting) off the appropriate number of pins from a 36-pin header, as shown in Figure 4.

Connect one of the end pins to a length of wire. Add a bit of 1/16" heat

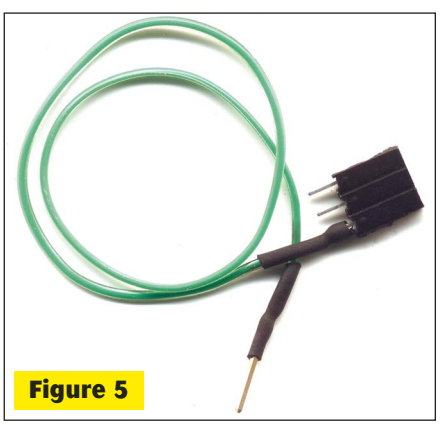


Figure 5

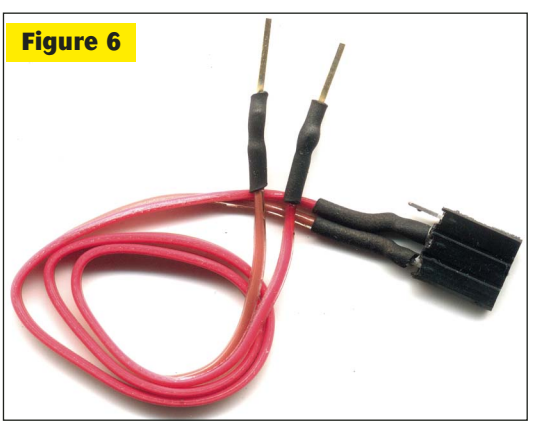


Figure 6

GOING FURTHER

I wanted to take the PC test program a step further. In the program RCRadioTestForm.txt, I display the actual joystick positions as shown in Figure 1. This program requires ZeusPro, but I have also included a compiled .exe of the program, as well.

In the August issue of *SERVO Magazine*, there was an article called FaceWalker. FaceWalker used a DiosPro to interface to a wireless Playstation controller. It would be a very simple task to replace that interface with the one presented here. Simply replace the UART interface to one that tests port 14.

shrink as reinforcement. At the other end of the wire, attach a single pin pulled from a male header, then add some heat shrink to that as shown in Figure 5. You will need one of these for each channel on the radio you are going to connect to the DiosPro.

If your receiver has a separate battery connector, you will need to attach two leads to a header as shown in Figure 6. Note than some receivers use BEC only and you must connect the power to one of the servo connectors. Refer to your receiver's user manual.

Once your connectors have been created, it is a simple matter to connect the appropriate connector to the Dios or Dios Carrier. You will need to refer to you receiver's manual to determine the appropriate connector. You can also look at one of the servos that came with your radio. The red wire is +, the black wire is –, and the white wire is the signal lead for the servo. On this Futaba radio shown here, there is a small key slot for each connector. The key is located next to the signal lead.

TESTING WITH A MICROCONTROLLER

To test the interface with a microcontroller, I will use another Dios and a program called ReadRCRadioTest.txt shown in Program 1. The program simply opens a port, sends the request command, reads the six channels, and displays the results. The output is shown in Figure 7.

Notice the tight loop. This is where you would actually process the received data and use it to control your world. You might be asking, "Why not just place any processing in the same chip that is measuring the pulses in the first place?" The answer is quite simple. The DiosPro runs at 40 MHz and at 10 MIPS, and we need almost 100% of this to accurately measure the six pulses from the receiver. By offloading the processing of the receiver data, we not only get accurate readings, we also divide and conquer. Once the interface chip has been debugged, we no longer have to worry about any of the complexities of this RC radio interface and can concentrate solely on our application. It's the hardware equivalent to modular programming.

Notice that just after we have opened the port we send the command:

PROGRAM 2

```
func main()
   dim x as integer


   x=ComOpen(1,baud=115200,port=1)
   print "Open Status = ";x
   ComDTR(1,0)
   pause(50)
   ComSettings(1,Timeout = 1,Priority = 10)
   'ComBGSuspend(1,0)

   pause(50)
   'Tell RCRadio Chip we want 6 channels
   ComOutput(1,chr(6))
   pause(50)


Loop:
  getreadings()
  pause(10)
  goto Loop


endfunc


'____________________
' Get 6 readings
'____________________
func getreadings()

  dim a,b,c,d,e,f,cmdidx
  ComPurge(1)
  doevents()
  ComOutput(1,chr(128))


  cmdidx = -1
Loop:
  ComGetPacket(1,Loop,Timeout,50,cmdidx,a,b,c,d,e,f)
  print a," ",b," ",c," ",d," ",e," ",f

  exit()

TimeOut:
  print "TimeOut"
  exit()
endfunc
```

PROGRAM 1

```
DiosPro
func main()

  hsersetup baud,HBAUD115200,txon,start,inwait,25
  pause 50
  hserout 6  'Set the number of channels
  pause 50

Loop:
  getreadings()
  pause 100
  goto Loop

endfunc

func getreadings()

  dim a,b,c,d,e,f
  hsersetup clear
  hserout 128

  hserin timeout,a,b,c,d,e,f
  print a," ",b," ",c," ",d," ",e," ",f

  exit

timeout:
  print "Timeout"
endfunc
```

```
hserout 6
```

This tells the radio interface that we want all six channels. If you only want two or four channels, you will need to change this command, as well as the hserin command.

One thing to keep in mind is that if you request six channels, you must have six channels connected to the radio. If a requested channel has no data, then no data will be transmitted until signal returns. For this reason, the host program should test for a timeout. If no data is received after 20 ms, then the transmitter is off or a connection is broken.

For the best reliability, you should tie the unused channels to Vss.

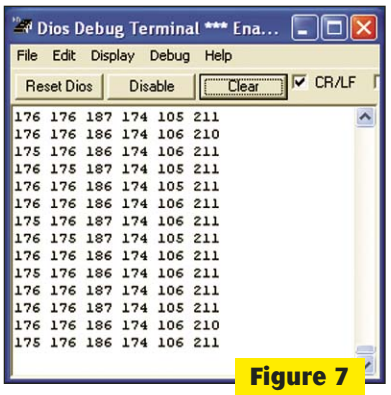


Figure 7

TESTING WITH A PC

In order to connect a PC or Pocket PC to the interface, you will need an RS-232 driver chip. The Dios Workboards all have built-in RS-232 drivers that can be used for programming or as an interface. Refer to the Dios Workboard manuals for setup instructions.

The program called RCRadioTest.txt shown in Program 2 is a Zeus program. This program will work with the free version of ZeusLite. Notice that this program, while not identical, is very similar to the Dios program presented earlier. SV

SOURCES

I have provided the part numbers for the carrier PCBs, as well as the EZ232 PCB. This way, you can build your own kits. For a list of the parts needed to build your own kits, just download the manuals for the appropriate board. These can be found on the website.

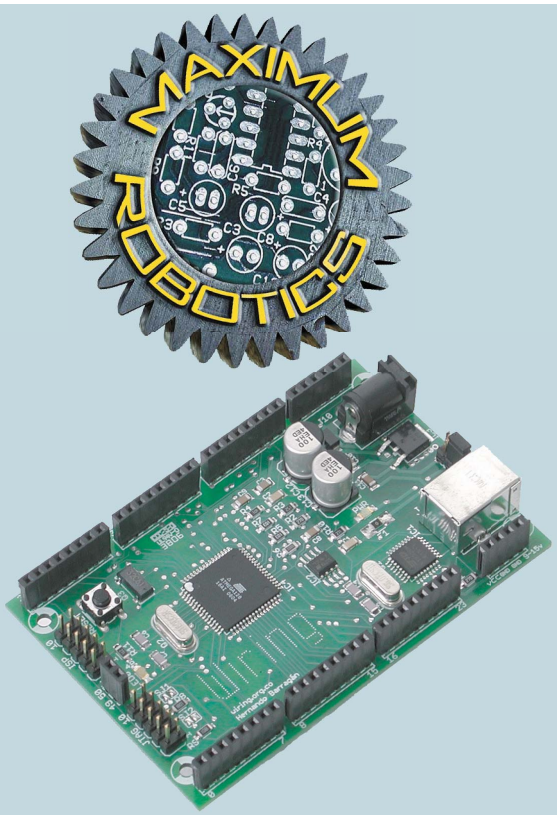
KRMicros — www.krmicros.com

- ZeusLite: *www.krmicros.com/Development/ZeusLite/ZeusLite.htm*

Kronos Robotics — www.kronosrobotics.com

• DiosPro28 Chip	#16169
• DiosPro40 Chip	#16168
• DiosCarrier1	#16170
• Carrier1 PCB	#16151
• Dios Workboard Basic	#16453
• Dios Workboard Deluxe	#16452
• Dios Workboard PCB	#16454
• 36-Pin Female Header	#16291
• 40-Pin Male Header	#16290
• EZRS232 Driver	#16167
• EZRS232 PCB	#16149

- Dios Compiler: *www.kronosrobotics.com/downloads/DiosSetup.exe*



Wiring Robot Controller

· 43 Digital I/O Pins
· 8 Analog Inputs
· 8 External Interupts
· 6 PWM Channels
· 2 Serial Ports including Bi-Directional USB
· 128k Memory
· The Wiring Programming Language

Grand Opening Sale! 15% off your first order!*

The Wiring language provides a simplified subset of C or C++ that hides more advanced concepts like classes, objects, pointers (while still making them accessible for advanced users). You get the power of C or C++ with the ease of a language like Basic. Programs written in Wiring are translated and then run as C++ programs on the Wiring I/O board.

~~$69.95~~ Only $59.46 after discount!

*Use code go15percent. Offer expires 12/15/2006

1-800-979-9130
MAXIMUMROBOTICS.COM

Give your Robot the power it deserves...
Introducing the
proSeed microcontroller

* 16 general purpose I/O's
* 16 analog inputs
* HUGE 128KB RAM
* 128KB EEPROM
* Super-Fast 70MHz SX CPU
* 24 pin compatible
* Programming languages
  16-bit ANS FORTH.
  C and Assembly available by third party.
* Visit our website for more...

LORAX WORKS
www.loraxworks.com

ALL ELECTRONICS CORPORATION
THOUSANDS OF ELECTRONIC PARTS AND SUPPLIES
VISIT OUR ONLINE STORE AT www.allelectronics.com

WALL TRANSFORMERS, ALARMS, FUSES, CABLE TIES, RELAYS, OPTO ELECTRONICS, KNOBS, VIDEO ACCESSORIES, SIRENS, SOLDER ACCESSORIES, MOTORS, DIODES, HEAT SINKS, CAPACITORS, CHOKES, TOOLS, FASTENERS, TERMINAL STRIPS, CRIMP CONNECTORS, L.E.D.S., DISPLAYS, FANS, BREAD-BOARDS, RESISTORS, SOLAR CELLS, BUZZERS, BATTERIES, MAGNETS, CAMERAS, DC-DC CONVERTERS, HEADPHONES, LAMPS, PANEL METERS, SWITCHES, SPEAKERS, PELTIER DEVICES, and much more....

ORDER TOLL FREE
1-800-826-5432
Ask for our FREE 96 page catalog

It's the most wonderful time of the year...

TO START YOUR SUBSCRIPTION TO SERVO MAGAZINE!
Call us at 877.525.2539
or go online at
www.servomagazine.com

At Last! Low cost encoders for motors & servos!

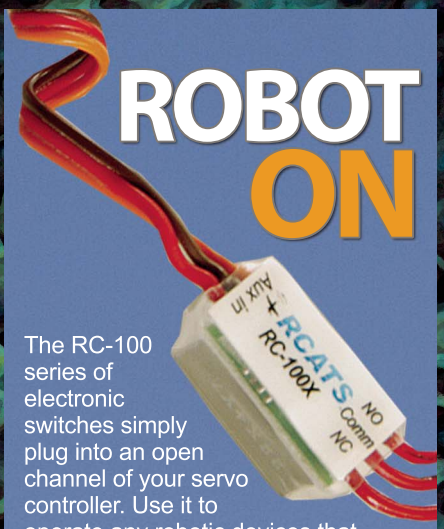
WheelWatcher™

- Easy installation
- Models compatible with Futaba, GWS, Hitec & Hobbico servos and many Solarbotics motors
- Uses standard wheels
- Dead reckoning, odometry, stall detection, closed-loop control

NU-BOTICS™

www.nubotics.com

ROBOT ON



The RC-100 series of electronic switches simply plug into an open channel of your servo controller. Use it to operate any robotic devices that require a simple on-off.

Capable of high currents up to 10A at 120V AC or 30V DC, and weighing in at 12 grams, turning on new opportunities are endless.

RCATS™

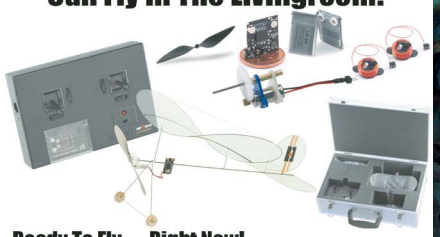
408-830-0745 | 3697 Enochs St, Santa Clara CA 95051

www.rcatsystems.com/servo

PLANTRACO MICRO R/C

CARBON BUTTERFLY

3 Gram Briefcase Airplane You Can Fly In The Livingroom!



Ready To Fly . . . Right Now!

The Carbon Butterfly ships to you in it's own aluminum briefcase and is ready for indoor flying action at a moments notice. Fly in any room 12' X 14'. Full proportional control all-digital R/C system and many innovations make this dream a reality!

Seeing is Believing - Visit our Website

Plantraco Ltd.
1105 8th Street E
Saskatoon, SK
Canada, S7H 0S3
306-955-1836 orders
306-931-0055 fax
www.plantraco.com

WWW.PLANTRACO.COM

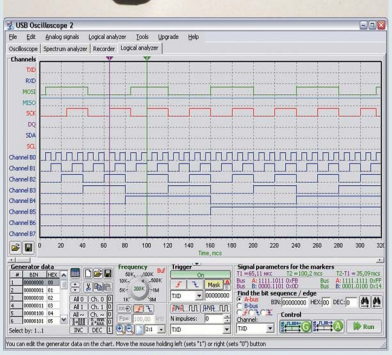
306-955-1836

USB Oscilloscope for $169.50

8+8 digital + 2 analog channels

Oscilloscope - Generator - Logic Analyzer with UART, SPI, I2C and 1-Wire interfaces

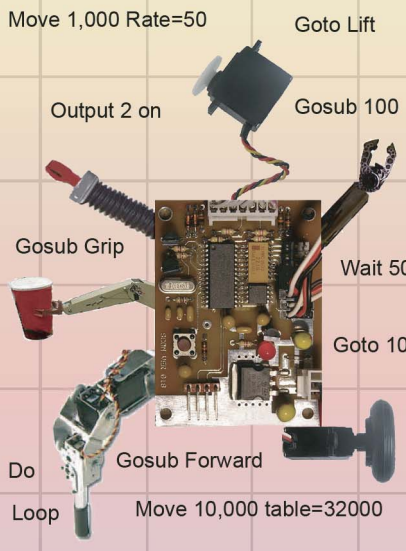
Great for PIC and AVR projects!



www.HobbyLab.us

Robotics Showcase

Scon is Servo Control !



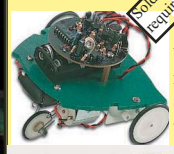
* Over 16 Basic-like Scon instructions
* 8 High resolution PWM servo outputs
* 256 KB flash memory / 32,000 elements
* Two Digital outputs & three inputs
* Extremely flexible rate & speed control
* Powerful multi-servo position manager
* Serial I/O port for setup & programming
* Serial slave operation capable

For further information and to order:

WWW.SCONCON.COM

The **Escape Robot**'s built-in microprocessor enables it to "think" on its own. (KSR4) **$29.95**

Soldering required



The robot frog moves forward when it detects sound and repeats: start (move forward) -> stop -> left turn -> stop -> right turn -> stop. (KSR2) **$19.95**

5mm White water clear **LED** 3.5V 10,000 mcd (AB287) **$0.56**

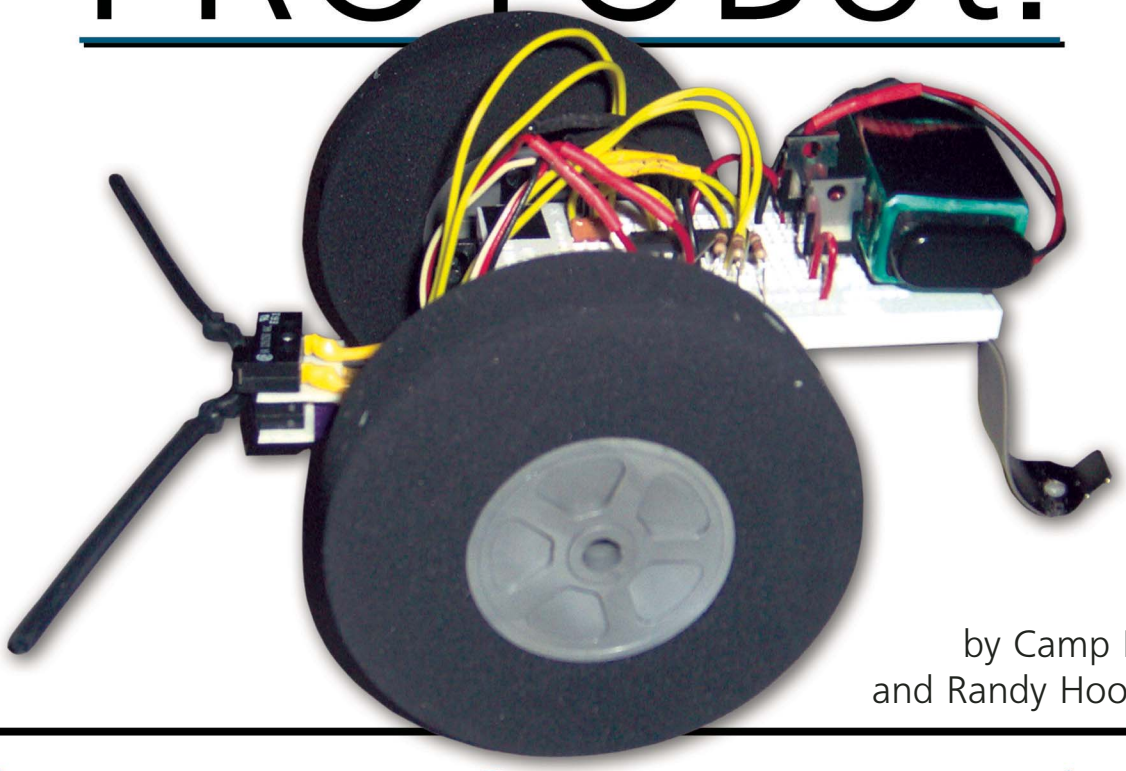
20 second voice recorder/playback module. The electret microphone is on the board. One button records, the other button is momentarily pressed to replay the message. (pre-assembled) (A96010) **$6.60**

The Velleman **Personal Scope** is not a graphical multimeter but a complete portable oscilloscope at the size and cost of a good multimeter. (HPS10) **$129.**

Thousands more items on our web site!

Electronics123.com
Tel: 1-888-549-3749
102 E. Park Ave, Columbiana OH 44408

PROTOBot:



by Camp Peavy
and Randy Hootman

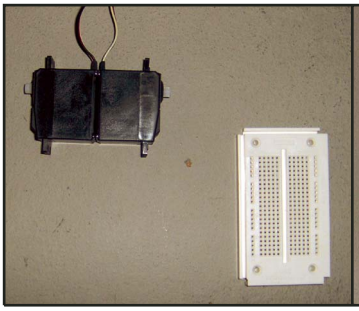
AMOEBA!

A Complete Interactive Robot

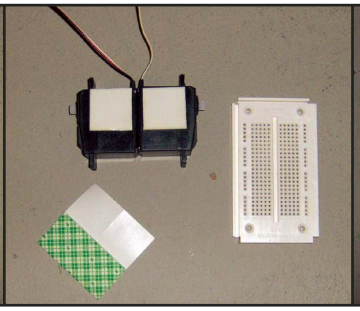
The basic concept behind the PROTOBot is that of a solderless breadboard on wheels. Any breadboard will do, but I like the "Experimentor 350" from Global Specialties; they're sold through RadioShack as #276-175. (A bigger one will allow you to add more sensors later.) After the solderless breadboard, the next major component is the continuous rotation hobby servo. Hobby servos (as opposed to industrial servos) are generally limited to 180 degrees of motion; for the rest of this article, we will use the term "servo" to mean "hobby servo." Originally, you had to modify the standard servo for continuous rotation; which is a good exercise if you haven't done it before or if standard servos are all you have. There are plenty of websites that describe this process; just Google "modify servo." Otherwise, Parallax sells continuous rotation servos as #900-00008.

Last year, I wrote an article for *SERVO Magazine* called "Introducing the PROTOBot" (October '05). In that article, I described how to build an inexpensive Stamp-based educational robot. I still believe the Stamp microcontroller from Parallax is one of the best ways to get started in robotics and suitable for more advanced applications, as well. However, Stamps cost approximately $50, which might be cost-prohibitive for some hobbyists.

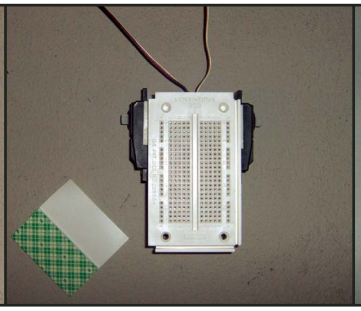
Enter the Ameoba! The idea behind Amoeba is to offer a complete interactive robot kit for $75 that teaches basic electromechanical techniques and can be upgraded to the BASIC Stamp II (BS2) or any other processor at a later time. The original Stamp-based PROTOBot costs ~$125. The cost of the PIC-based Amoeba is ~$50; due mainly to the price reduction of the microcontroller. The advantages of the BASIC Stamp are the PBASIC development environment is free and the Stamp is easy to program in circuit. Ease of use means fun and entertainment ... which



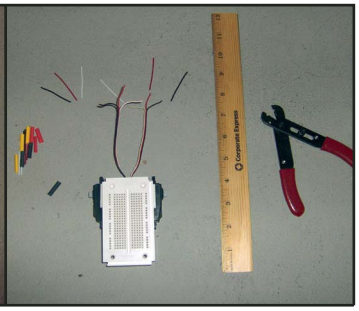
STEP 1: Lay the continuous rotation servos back-to-back with gears facing outward.



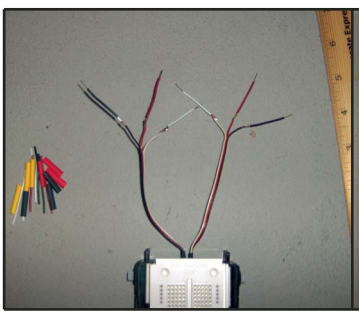
STEP 2: Put a square of double-stick foam tape towards the front and center of the two servos.



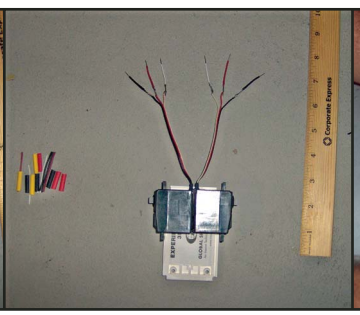
STEP 3: Carefully stick the solderless breadboard to the two servos aligning the servos with the front edge; press firmly.



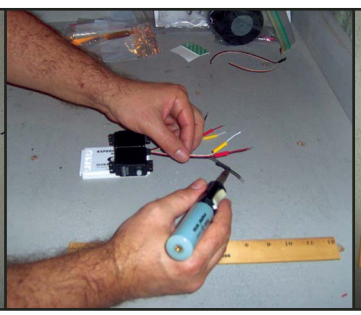
STEP 4: Cut the servo leads to 4" and strip 1/4" insulation off the ends; spread the wires apart. We'll be soldering 22-gauge solid-core wire to the ends creating breadboard "plugs."



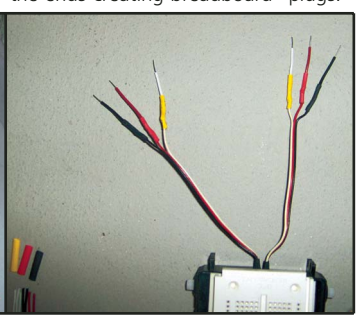
STEP 5: Twist the stranded servo wire around one end of the 1.5" 22-gauge solid-core wire with 1/4" stripped off each end. Consider doing the wires one at a time.



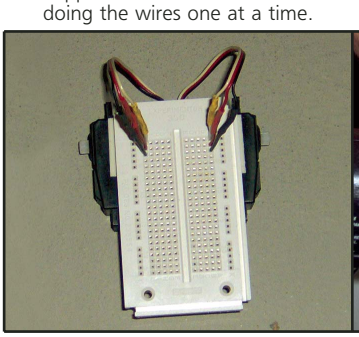
STEP 6: Solder the six servo leads.



STEP 7: Cover the solder joints with heat shrink tubing; otherwise, the wires will short. If you don't have heat shrink tubing, use electrical tape or duct tape.



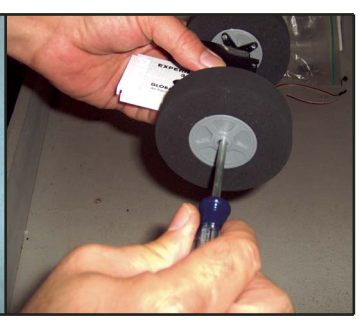
STEP 8: You should end up with six breadboard pins: one positive, one negative, and one signal on each servo.



STEP 9: Plug the servo wires into the solderless breadboard; this is just temporary until we wire up the rest of the system.

STEP 10: Prepare the wheels. Drill out a center bore to 1/4". If you don't do this, the servo screw will not fit!

STEP 11: Drill four small holes (1/16") for the servo horn. Use the servo horn as a template; this makes it much easier to thread the screw. If you don't have a 1/16" drill bit, the screws can be driven manually.



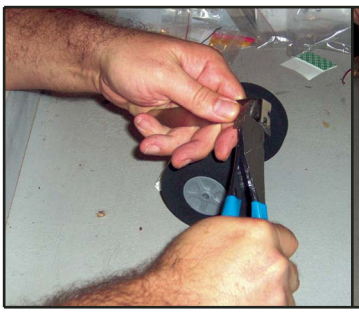
STEP 12: Attach wheel to servo with servo screw.

makes you more apt to work with it. The PIC, on the other hand, requires an IC programmer (the EPIC programmer from microEngineering, for example) and unless you're fluent in assembly language and/or C, the software isn't free; $90 for PICBasic and $250 for PICBasic Pro. (Both are from microEngineering Labs at **www.microengineeringlabs.com**.) The good news is one can actually get an easy-to-use Basic language compiler for the PIC chips and microEngineering Labs does have a free trial download version of PICBasic Pro; the limitation is 31 lines of code. Fortunately, the code that goes into the basic Amoeba actions takes only 25 lines and could be compressed further. Again, if you want to burn your own chips for the Amoeba, you'll need a programmer (EPIC ~$60). Amoeba kits and pre-programmed PIC16F628 chips are available through **www.camppeavy.com**.

Another Amoebic discovery ... double-stick foam squares (Ace #91644 — $3.49) designed to hang up to two pounds on the wall (pictures, decorations, etc.). In the previous article, I waxed poetic about the virtues of an adhesive called "E60000." I still like adhesives, but foam double-stick tape has great functionality because there's literally no wait time and the hold-strength for tabletop robots is good enough.

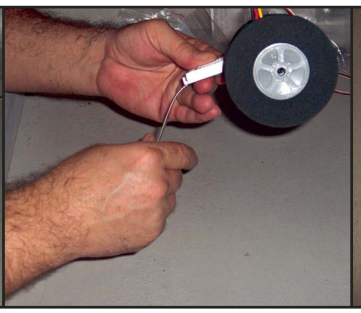
We begin this process by building the body (Steps 1-18). First, lay the continuous rotation servos back-to-back with the gears facing outward (Step 1). Much of Amoeba's construction involves simply soldering 22-gauge solid wire onto stranded wire thereby making breadboard "plugs" which can then be interfaced with the microcontroller.

Stick the protoboard to the two servos (Steps 2-3). Cut the two servo leads to 4" and strip 1/4" insulation

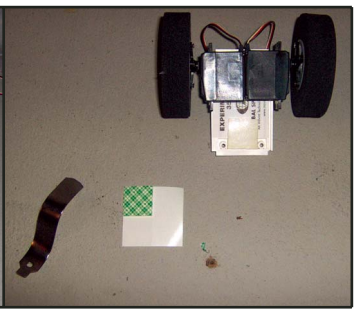


STEP 13: Cut the right-angle top from the PC-slot cover. You will have to use a pair of "dikes" (diagonal cutters); you may have to bend it back and forth to break it.

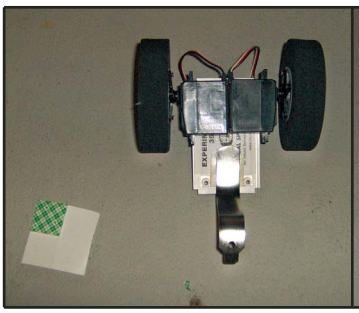
STEP 14: Trim it up so it looks nice.



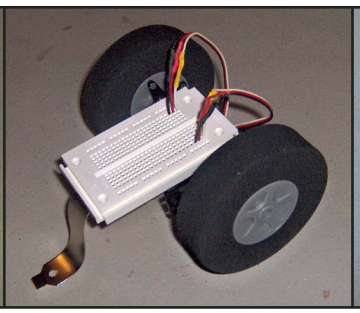
STEP 15: Shape the PC-slot cover so it's the proper height and angle for a tail skid.



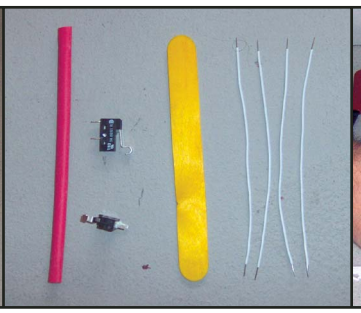
STEP 16: Put a double-stick foam square on the bottom of the solderless breadboard where you want to mount the tail skid.



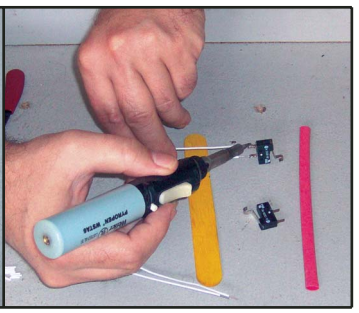
STEP 17: Mount the tail skid.



STEP 18: Plug the servo pins into any of the tie-points or "holes" in the breadboard. The body is now ready for action! At this point, you could use whatever processor you like.



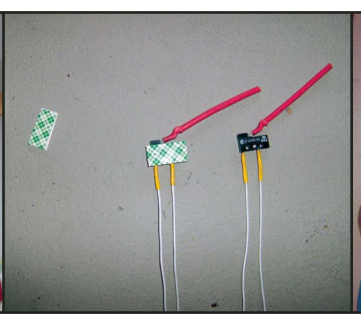
STEP 19: Cut four lengths of 22-gauge solid-core wire to 6" and strip 1/4" insulation from each end.



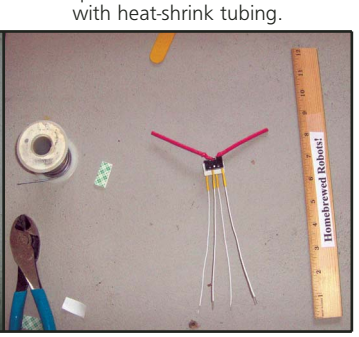
STEP 20: Solder one end of the 6" 22-gauge solid-core wire to each terminal on the #E61 snap-action switch and cover with heat-shrink tubing.

STEP 21: Cut two lengths of 3/16" heat shrink tubing for "whiskers."

STEP 22: Shrink the 3" whiskers over the snap-action lever. If you don't have a heat shrink gun or blow-dryer, you can shrink the tubing with a soldering iron.



STEP 23: Cut a double-stick foam square in half and place it on one of the whisker-switches; be sure and not allow the tape to affect the "snap-action."



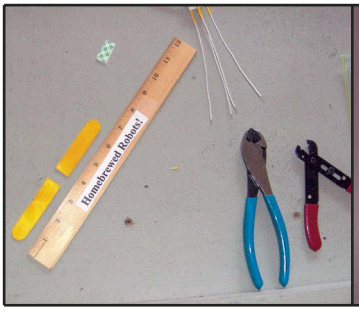
STEP 24: Carefully place one switch on top of the other facing the opposite direction so the levers go out radially like whiskers; when straight, press firmly.

off the end (Step 4). Wrap the three stranded wires from each servo around the end of one 1.5" piece of 22-gauge solid wire (Step 5). Solder the six wires (Step 6). Cover the solder joints with heat shrink tubing or electrical tape (Step 7). You should end up with six breadboard "pins" that you can plug into the protoboard (Step 8). Use this technique with any sensor or output device. Just solder 22-gauge solid wire and plug it in! Plug the Amoeba's servo pins into its body; anywhere for now as this is a temporary place to plug it in until we wire up the rest of the system (Step 9).

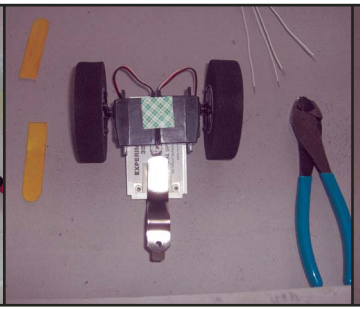
In prototyping, quickness results in productivity because you are able to try more things. In the original PROTOBot article, we glued the servo horns; this time, we screw them in because it is quicker (#2 x 1/2" wood screws) (Step 11); but before you do that, you'll want to drill out the center 1/4" or you will not be able to put the servo screw through (Step 10). Put the wheel on the servo and screw it in (Step 12).

For the original PROTOBot, I used a nice tail wheel which I still suggest for improved mobility but it costs $10. For the Amoeba, we've gotten more practical and will use a PC slot cover (Jameco #11754 — $1.79). You probably have a half-dozen or so in the old computer in the garage. These are the "covers" that you would remove as you added in PC cards to the computer. To start with, cut off the end of the bracket with the 90 degree angle (Step 13). You'll need a pair of diagonal cutters. This flat piece of metal can be used for the tail skid (Step 15). Stick the tail skid to the solderless breadboard with a double-stick foam square like you did with the servos (Step 17). Shape it so it balances the robot. Now that you've completed the Amoeba body ... it's time for the guts (Mwa-ha-haaa!), er electronics.

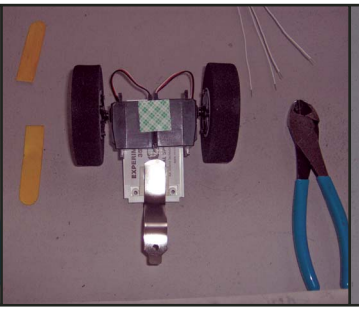
In PROTOBot, I used a tri-shaped 2.75" wide bumper made from matt



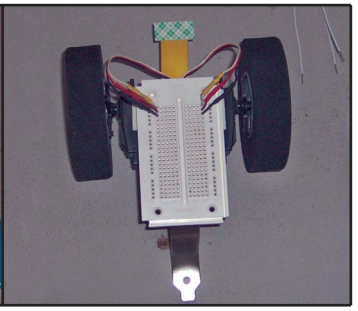
STEP 25: Cut a jumbo craft stick to a length of 3".



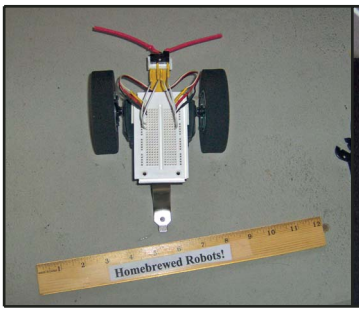
STEP 26: Stick a square of double-stick foam tape to the front and center of the two servos.



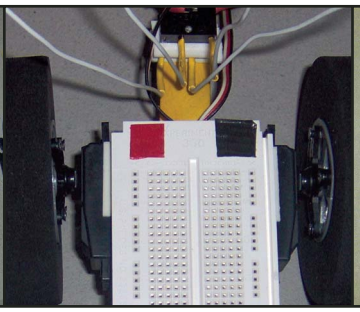
STEP 27: Carefully position the 3" cut craft stick to be centered and protrude 2" beyond the front of the servos; press firmly.



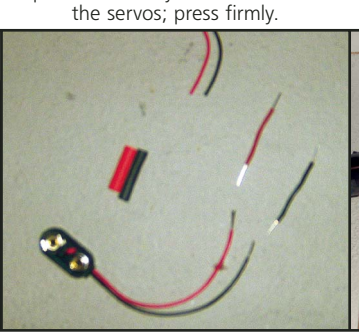
STEP 28: Stick a 1/2" double-stick foam square on the end of the jumbo craft stick.



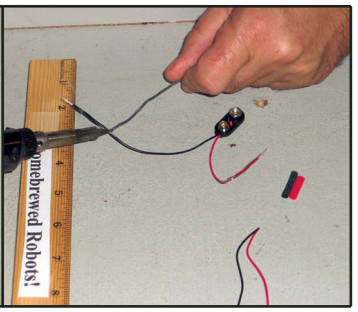
STEP 29: Stick the whisker array on the end of the craft stick and plug the wires into the protoboard ... anywhere, for now.



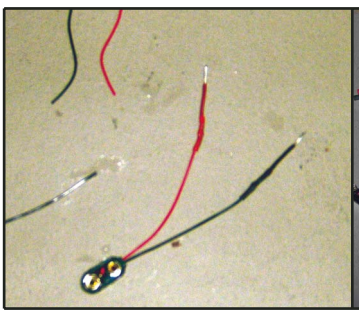
STEP 30: Place red tape on one rail and black tape on the other. If you don't have tape, a red and black marker will do the job. At the least, mark the negative rail with a black mark.



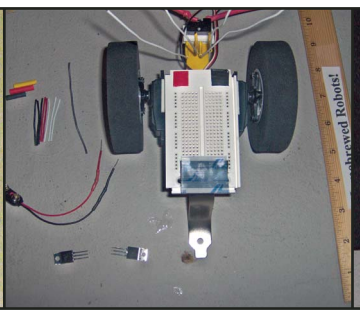
STEP 31: Next, we'll build the power cable. Cut the leads on the nine-volt battery snap to 3" and cut two pieces of 1.5" solid-core wire to 1.5"; one red, one black. Strip the ends of the solid-core wire 1/4".



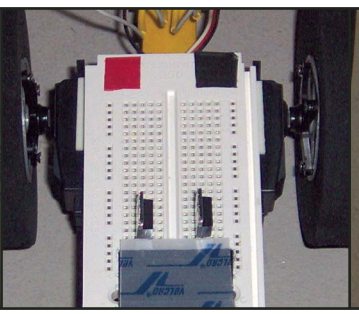
STEP 32: Solder the red solid-core extension to the end of the red power lead and the black solid-core extension to the black power lead.



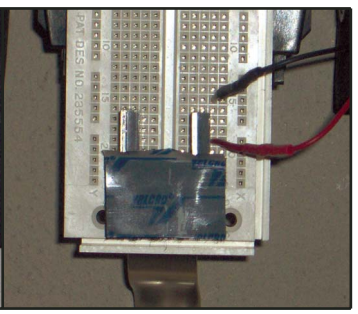
STEP 33: You'll need to put heat-shrink tubing around the solder joint, otherwise the two conductors will touch and cause a short circuit (not good!).



STEP 34: Put one side of the adhesive Velcro perpendicular and to the rear of the solderless breadboard. The other side will adhere to the nine-volt battery.



STEP 35: Place the two 7805 voltage regulators in front of the Velcro on opposite sides of the breadboard's center gap. Notice the orientation of the chips with heatsinks facing left. Be aware the heatsinks will get hot!



STEP 36: Now plug the positive (red) wire from your nine-volt battery snap into pin 1 on the right-hand 7805. For now, plug the negative lead (black) into a hole two or three above the 7805.

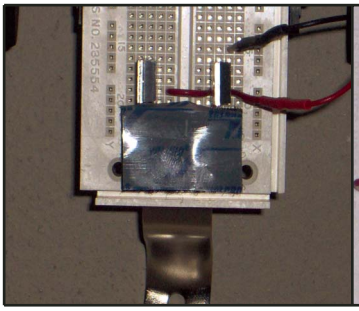
board. This is actually a better design because it is broader, but for the Amoeba, we've chosen simplicity over functionality. I've simplified the design into two 3" long whiskers. The idea is to get you to an interactive robot as quickly as possible.

Steps 19-29 are concerned with the whisker sensors, Steps 30-38 the power system, and Steps 39-45 the brain. After building the body and installing the whiskers, it is time to set up the power system. As we delve into the electronics of the system, it is important to understand how solderless breadboards actually work. Usually protoboards have rails along the sides (columns) which carry positive (+) and negative (-) connections and perpendicular rows of five tie-points or "holes" isolated from each other by a 1/16" gap. This gap is usually straddled by a DIP (Dual Inline Package) chip such as the PIC or the Stamp. One should understand solderless breadboards and electronic diagrams as you take on the rest of the Amoeba project. If you've never used a solderless breadboard, you should research the subject before proceeding as understanding the basic layout of the solderless breadboard is critical.

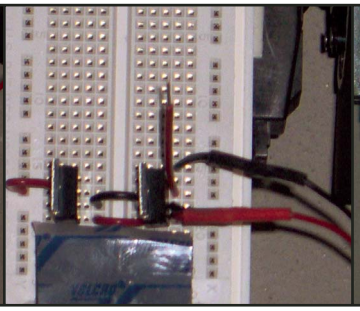
The original PROTOBot article included a brief description or you can Google "solderless breadboard" to yield numerous useful results.

The first thing I like to do as I set up the power system is to place red tape on one rail and black tape on the other. If you don't have tape, a red and black marker will do the job. At the least, mark the negative rail with a black mark (Step 30).

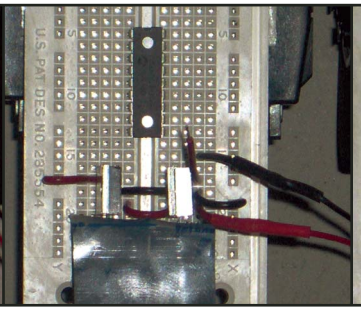
Next, we'll build the power cable. Cut the leads on the nine-volt battery snap to 3" and cut two pieces of 22-gauge solid wire to 1.5"; one red, one black. Strip the ends of the solid wire 1/4" (Step 31). As with the servos and whiskers, much of the



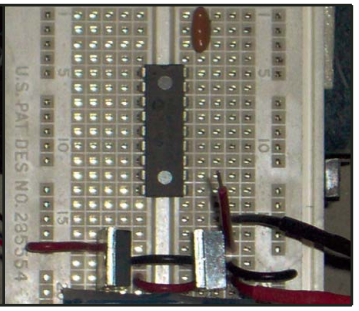
STEP 37: Jumper pin 1 of the 7805 on the right to pin 1 of the 7805 on the left with a 1.5" (preferably red) piece of 22-gauge solid-core wire stripped 1/4" off each end.



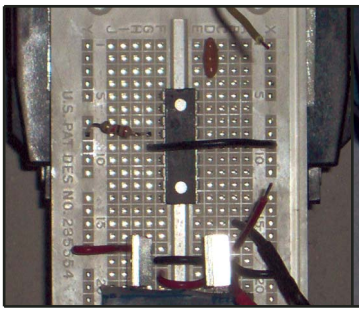
STEP 38: Connect the center pins (pin 2) on the two 7805s, then connect the center pin on the 7805 on the left to your negative rail which we've marked with black tape on the left side.



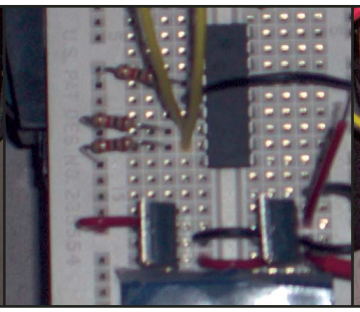
STEP 39: Take the PIC16F628; orient it with the notch up on the package. Pin 1 should be up and to the left.



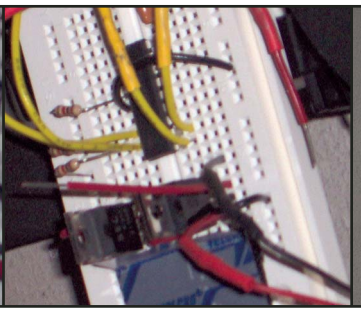
STEP 40: Plug the 4 MHz resonator into the first three rows of the solderless breadboard.



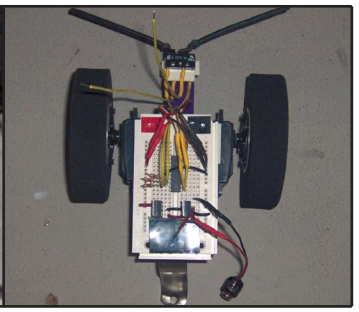
STEP 41: Pin 4 — 10K ohm resistor to +5 volts; pin 5 — ground (2" 22-gauge solid wire; black; stripped 1/4" at each end).



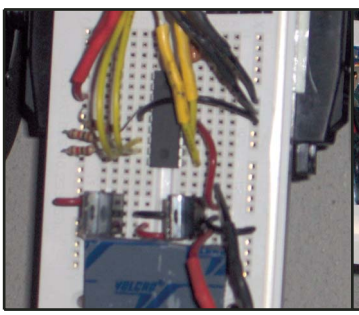
STEP 42: Pin 7 — 10K ohm resistor to +5 volts and one wire from the right snap-action switch; pin 8 — 10K ohm resistor to +5 volts and one wire from the left snap-action switch.



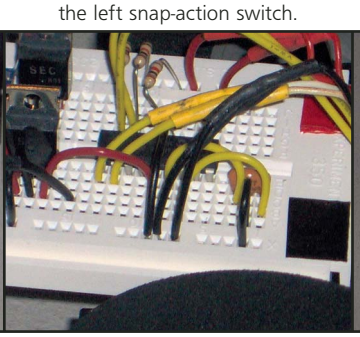
STEP 43: Pin 10 — white wire from left servo; pin 11 — white wire from right servo.



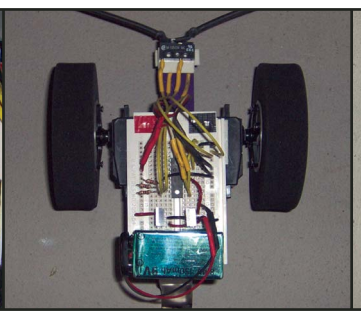
STEP 44: Connect the two negative or black wires from the servos to the negative rail (black). Connect the two positive or red wires from the servos to the positive rail (red).



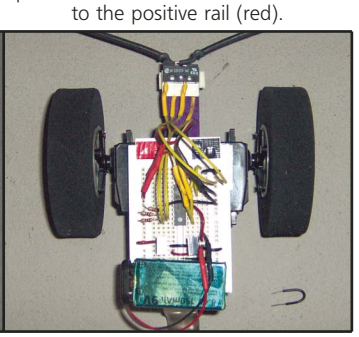
STEP 45: Pin 14 — output (red wire) from right-hand 7805.



STEP 46: Pin 15 — connect to pin 1 of resonator; pin 16 — connect to pin 3 of resonator. Connect center pin of resonator to the ground rail. Connect the two remaining wires from the snap-action switches (whiskers) to the negative rail.



STEP 47: Attach other side of adhesive Velcro to nine-volt battery; press onto solderless breadboard and attach battery snap. Nothing should happen yet.



STEP 48: Plug in the wire switch between the nine-volt black wire and the negative rail (look at the schematic). Your robot should spring to life; to turn him off, remove the wire. Later, you might want to get a real SPST switch; glue or tape it to the side.

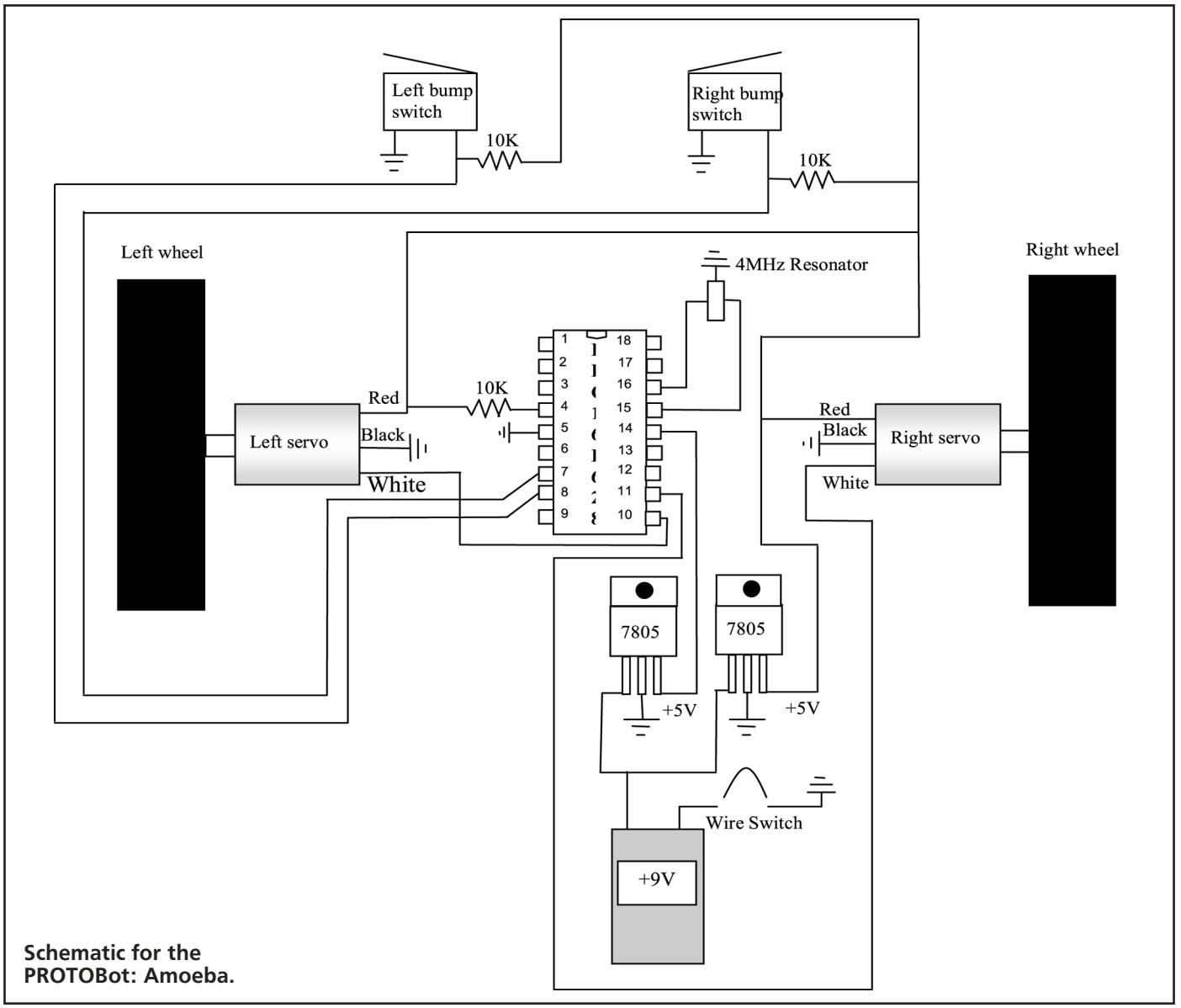
construction consists of soldering 22-gauge solid wire to whatever sensor or actuator you want to include in the system. Solderless breadboards are good enough for many hobbyist applications.

Solder the red solid extension to the end of the red power lead and the black solid extension to the black power lead (Step 32). The color, of course, doesn't really matter but as long as you're taking time to do it, you may as well observe the color codes. Besides, it makes it more educational and easier to troubleshoot in the future.

You'll need to put heat shrink tubing around the solder joint, otherwise the two conductors will touch and cause a short circuit (not good!). If you don't have a heat-shrink gun or hair blow-dryer, you can shrink the heat shrink tubing with your soldering iron. If you don't have heat shrink tubing, use electrical tape or duct tape (Step 33).

Put one side of the adhesive Velcro perpendicular and to the rear of the solderless breadboard (Step 34). The other side will adhere to the nine-volt battery. You'll get ~30 minutes from one battery so if you plan on experimenting with the Amoeba for an extended length of time — or robotics in general — consider investing in rechargeable batteries and a charger. There are two voltage regulators in the system ... the popular "7805." They look like three-legged black Chicklets. Place the two 7805 voltage regulators in front of the Velcro on opposite sides of the breadboard's center gap. Notice the orientation of the chips with heatsinks facing left (Step 35).

Now, plug the positive (red) wire from your nine-volt battery snap into pin 1 on the 7805 on the right (Step



Schematic for the PROTOBot: Amoeba.

36). Pin 1 is the 7805's input. Pin 2 is ground and pin 3 is the output ... in this case, five volts; just right for driving servos and running microcontrollers like the PIC! Normally one would use only one 7805 voltage regulator but the servos generated so much electronic noise that I had to split the power supply into two separate regulators: one for the servos and one for the microcontroller. Amazing things these 7805s ... put in between seven and 35 volts (like our nine-volt battery) and out comes five volts regulated! This means that the voltage will stay consistent (+5 volts) regardless of the current draw: be aware the heatsink on the voltage regulators will get hot! After connecting the positive lead of your battery, connect the negative lead. Here, I want you to plug it into an empty row just above the 7805 (Step 36). The last thing we'll do is add a wire switch which connects the battery's negative terminal to ground and your robot will be "on" its way. To turn him off, remove the wire. Later you will want to get a real SPST (Single Pole Single Throw) switch.

Connect pin 1 of the 7805 on the right to pin 1 of the 7805 on the left with a 1.5" piece of 22-gauge solid wire stripped 1/4" off each end (Step

Pin Connections for the PIC16F628

- Pin 1 — No connection
- Pin 2 — No connection
- Pin 3 — No connection
- Pin 4 — 10K ohm resistor to +5 volts
- Pin 5 — Ground (2" 22-gauge solid-core wire; black; stripped 1/4" at each end)
- Pin 6 — No connection
- Pin 7 — 10K ohm resistor to +5 volts and one wire from the right snap-action switch
- Pin 8 — 10K ohm resistor to +5 volts and one wire from the left snap-action switch
- Pin 9 — No connection
- Pin 10 — White wire from left servo
- Pin 11 — White wire from right servo
- Pin 12 — No connection
- Pin 13 — No connection
- Pin 14 — Output (red wire) from right-hand 7805
- Pin 15 — Connect to pin 1 of resonator
- Pin 16 — Connect to pin 3 of resonator
- Pin 17 — No connection
- Pin 18 — No connection

Complete Parts List

THE BODY
- Solderless Breadboard: RadioShack #276-175 — $8.39
- Continuous Rotation Servos: Parallax #900-00008 (2 x $6.95) — $13.90
- Double-Stick Squares: ACE #91644 — $3.49
- 2 x 1/2" Wood Screws: ACE #1889 (8 x .09) — .18
- Drive Wheels: Dave Brown Products Lite Flite Wheels (pair) #WH35-5535 — $6.25
- Tail Skid: PC Slot Cover Jameco #11754 — $1.79

WHISKER SENSORS
- 3/16" Heat Shrink Tubing: Tech-Tron TT-100 3/16" RED (4') ($1.39)
- One Jumbo Craft Stick: **www.chenillekraft.com** or Haagen-Dazs — $1.95
- Two Snap-Action Switches (Cherry #E61-00R): Jameco #456382 (2 x $4.85) — $9.70

POWER SYSTEM
- Heavy Duty Nine-volt Battery Snap: RadioShack #270-324 — $2.59
- Two 7805 Voltage Regulators: RadioShack #276-1770 (2 x $1.59) — $3.38
- 1" Sticky Back Velcro (1 hook, 1 loop): ACE #5006036 — $1.50

THE BRAIN
- PIC16F628 Microcontroller: Jameco #193447 — $3.49
- Three 1K 1/4 Watt Resistors: RadioShack #271-1321 — 5 for .99
- One three-pin 4 MHz Ceramic Resonator: **www.glitchbuster.com** #CR-4 — .44

WIRES AND HEAT SHRINK TUBING
- 22-Gauge Solid Wire (five colors): Philmore Wire Kit #12-2276 — $5.99
- Black 1/8" Black Heat Shrink Tubing: Tech-Tron TT-100 1/8" BLK (4') — $1.39
- Red 1/8" Red Heat Shrink Tubing: Tech-Tron TT-100 1/8" RED (4') — $1.39
- Yellow 1/8" Yellow Heat Shrink Tubing: Tech-Tron TT-100 1/8" YEL (4') — $1.39

PROGRAMMER
- EPIC Plus — microEngineering Labs — $60 (**www.microengineeringlabs.com**)

SOFTWARE
- PICBasic Pro Compiler — microEngineering Labs — $249

37). Be sure the two 7805s are facing the same direction with their metal heatsinks to the left. The nine-volt positive connection (red wire) goes nowhere but the two pin 1s on the 7805s.

Connect the center pins (pin 2) on the two 7805s, then connect the center pin on the 7805 on the left to your negative rail which we've marked with black tape on the left side (Step 38).

The 7805 on the left will be powering the servos … put a 1.5" piece of 22-gauge solid wire (red) into pin 3 on the 7805 and the positive rail (the column of holes along the left side); it will supply five volts to the servos and switches. The 7805 on the right side will be dedicated to powering the PIC-chip; put a 2" piece of red 22-gauge solid wire into pin 3. Do not plug the other end into anything yet.

Doublecheck your wiring with the schematic shown here.

Take the PIC16F628; orient it with the notch up on the package. Pin 1 will be up and to the left. Plug the chip into the solderless breadboard (straddle the center gap; Step 39); leave at least three rows in the front for the 4 MHz resonator <queue sci-fi music>. No other wires should be connected to the rows at this time. Plug the 4 MHz resonator into the first three rows of the solderless breadboard (Step 40).

Okay, now we're going to work around the CPU (Steps 41-45). Refer to the sidebar "Pin Connections for the PIC16F628" and the schematic.

Connect the two negative or black wires from the servo to the negative rail. Connect the two positive or red wires from the servo to the positive rail (Step 44).

For Pin 15 — connect to pin 1 of the resonator (with 3" jumper wire); connect pin 16 to pin 3 of the resonator. Connect the center pin of the 4 MHz resonator to the ground rail (use 3" black jumper wire). Connect the two remaining wires from the snap-action switches (whiskers) to the negative rail (Step 46). Note that the microcontroller has been removed for programming.

Attach the other side of the adhesive Velcro to the nine-volt battery; press onto the solderless breadboard and attach the battery snap. Nothing should happen yet (Step 47).

Plug in the wire switch between the nine-volt black wire and the negative rail. Your robot should spring to life (Step 48). If nothing happens, unplug the wire switch and go back and doublecheck your wiring to the circuit diagram. To turn off your robot, remove the wire switch.

WRAP-UP

Is this the end or just the beginning? Consider replacing the pre-programmed PIC chip for an easy-to-program BASIC Stamp II. For information on how to do this, look for the first PROTOBot article at **www.camppeavy.com/articles/protobot.pdf** and the Ultimate TABLEBot at **www.camppeavy.com/articles/ultimate.pdf**. Or, order the back issues from the *SERVO Magazine* website at **www.servomagazine.com**.

The PROTOBot design gives a lot of flexibility on a low-cost platform. If you want to continue with the PIC, get the EPIC programmer and have at it! Robot building is a fascinating hobby and PROTOBots are a great way to get started. They're also a fun and easy project if you're an experienced builder. **SV**

by Dave Calkins

ROBOGAMES PREP

Firefighting Robots:

One of the Toughest Robot Challenges

Fourteen years ago, Dave Ahlgren of Trinity college started the Firefighting robot contest. The rules have evolved, but the goal remains the same: find a fire and put it out.

Unlike most robot contests — soccer, combat, magellan, sumo, etc. — firefighting robots have a real-world humanitarian goal: to save lives. You can now buy a robot that will vacuum your living room automatically. You go to bed, and while you're asleep, your living room is magically vacuumed. While this is certainly a time-saver, I don't think it will save any lives (unless the UPS guy is violently allergic to the dog hair on your rug).

Firefighting robots, on the other hand, can definitely save lives. What about smoke detectors, you ask? Well, smoke detectors only detect smoke — they don't attack the smoke or the fire causing it, they only alert you to it (presuming you remembered to change the batteries). But where there's smoke, there's fire. Wouldn't it be great if that fire never made enough smoke to set off the alarm?

Let's say you leave your laptop on the couch, and the battery decides to go kaboom after you've gone off to bed. When the smoke detector goes off, you run out of the house and wait for the firetrucks. But isn't it better if the firetruck is sitting in your home? A firefighting robot could be constantly charging and monitoring your home for smoke and flames. When it detects a fire, it immediately launches itself from its station, finds the fire, and extinguishes it.

Sure, you're going to have to replace the couch, but isn't it better to save the rest of the house? That's the goal of firefighting robots. Put out the fire before it gets too big. The people who perfect the firefighting robot are going to make millions. And thus, we have Trinity.

So, how do you build one? Well, first and foremost this is an autonomous robot. Like building a RoboMagellan bot, you've got your work cut out for you. You not only have to build a robust platform that can go up and down stairs and around tight corners, you've got to make it smart enough to go through multiple rooms, with sensors smart enough to avoid colliding with the walls and to tell where the fire is (in the case of the competition, a burning candle is used).

The candle is randomly placed in the corner, edge, or center of one of the scale rooms within the roofless "house." The house is a 2.43 meter square maze with four unique rooms separated by hallways (just like a real two-bedroom house). There is an "island room" in the house, so simple wall following won't get you into every room. The candle sits on a white sheet of paper, and your robot has to find the candle and blow it out while on the white area.

Your score is based on how fast you can complete the event, but you get three chances. The lowest score among all competitors is the winner. Penalty points are taken off your score for sliding along the walls and/or touching the candle.

Sound simple? It's not. The hardest problem is detecting the candle amidst all of the bright spots from ambient lighting. Most robots use a Hamamatsu UVTron fire detector (so much so that is should be counted as standard equipment). The Hamamatsu is the heart of a good firefighter. If you don't have a good fire detector, you're just wasting time. This will significantly reduce the frequency of false-positives.

But it cannot be the only sensor you use. Three-time gold medal winner Tony Pratkanis's robot Solenopsis invicta uses two Sharp GP2D12s to follow the walls, IR sensors and bumpers to navigate, and a line detector to figure out where the rooms are and where the white area under the candle is. All of this is controlled via the Grandar ASM robot base.

Start with a robust platform — Octobot from the Jameco Robot Store, the ASM from Grandar, or the 4WD1 from Lynxmotion are all excellent platforms, although any R/C truck would work. There's no sense in reinventing the wheel. Work on your software's ability to find the flame and put it out, then start expanding into

> “Fourteen years ago, Dave Ahlgren of Trinity college started the Firefighting robot contest. The rules have evolved, but the goal remains the same: find a fire and put it out.”

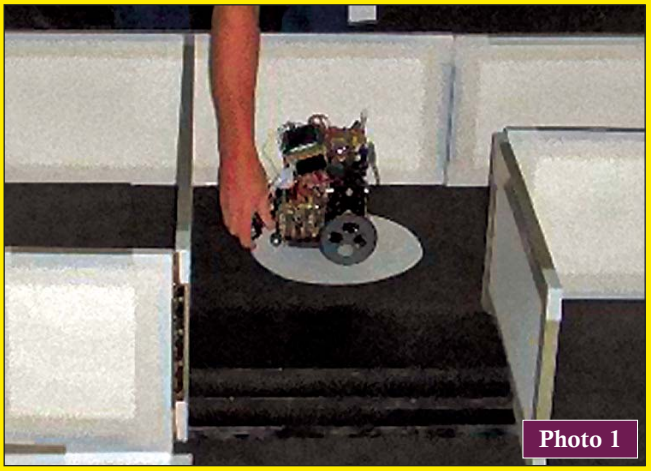


Photo 1

Photo 2

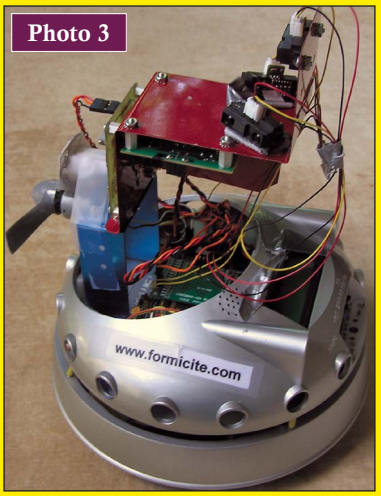


Photo 3

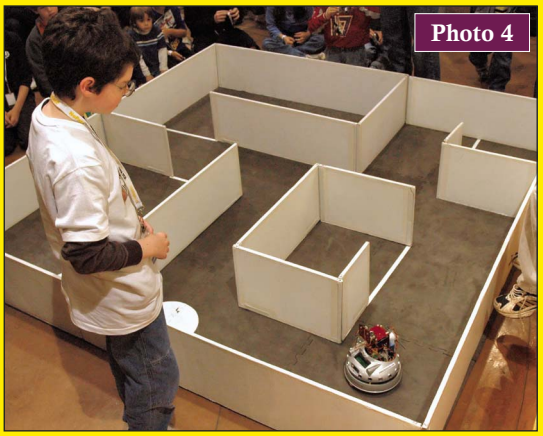


Photo 4

Photo 5

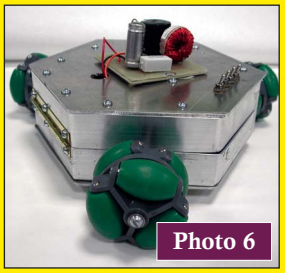


Photo 6

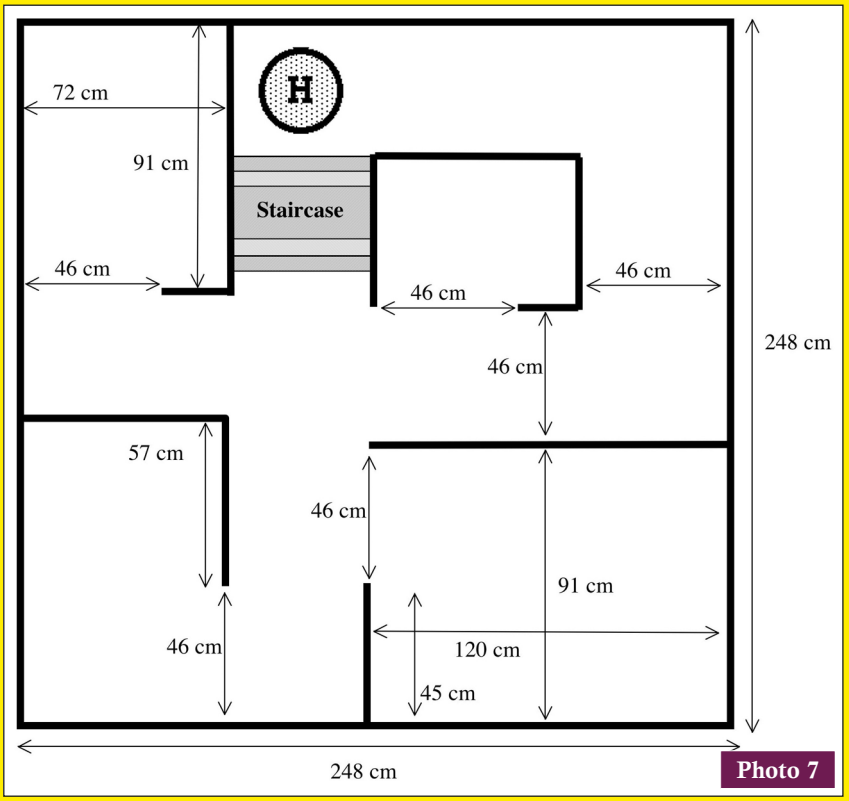


Photo 7

PHOTO 1. Going down stairs can be a problem for many robots.

PHOTO 2. Ted Larson looks on as his robot tries to find the candle.

PHOTO 3. Three-time gold medalist, Solenopsis invicta.

PHOTO 4. Three-time gold medal winner Tony Pratkanis watches his robot, Solenopsis invicta.

PHOTO 5. FlameOut went on to take the Bronze medal at ROBOlympics 2004.

PHOTO 6. Tryclops, an omni-wheeled firefighter.

PHOTO 7. The layout for the current Firefighting contest.

better or customized platforms.

Mark Whitney of Acroname has several tips for helping to ensure you do well. His full set of tips can be found at **http://trinity-fire.notlong.com**.

• *A multi-spectral approach for candle detection is quite effective*. Lighting conditions can easily spoof a system that relies solely on infrared or visible light. It's much harder to spoof a system that combines UV, visible, and IR sensors. In addition to UV-IR sensors, the Eltec Pyroelectric sensor provides true heat-sensing capability. However, it is a heat differential sensor, so it must sweep for best results. The signal processing for its analog output can be as simple as a window comparator circuit.

• *Use a fan if you want things to be simple and reliable*. Turn it on and blow out the candle. Water, CO_2, balloons, etc., are nice ... but when you use them up, you're toast. You can blow a fan as many times as you want, repositioning the robot if necessary until you put the flame out.

• *Do not underestimate the horrifically harsh lighting conditions at the contest*: blinding gym lights, moving shadows, candle reflections on bright white walls, flash bulbs reflecting on black floors, IR from camcorders and cameras, etc. You will use light for nearly everything — candle detection, obstacle detection, line detection. To test, turn on every light in your testing area, and then some. Try pointing your TV remote at your robot. Take some flash pictures while its moving. Bring in bright flashlights and make shadows. If your robot passes these tests and laughs at your futile attempts to fool him or her, then you have a chance at the contest.

• *If you are using a compass to help your robot navigate, be aware that the magnetic field can be distorted by nearby pipes and other metal objects*. In fact, compass readings can vary noticeably from one arena to the next at the contest site in Hartford.

• *Exclusive use of dead-reckoning (measuring the course and rogramming your robot to follow the measurements) for navigation will require you to make a full replica of the house for serious testing*.

• *Test your candle detection and candle extinguishing systems at the minimum and maximum possible heights of the candle* (15 cm to 20 cm off the floor). Test them with your robot as close to the candle as it can get. Test them at the maximum legal extinguishing range (30 cm). Make sure it can blow out the candle at all extremes.

• *As batteries are used up, motors slow down and robot timing changes*. In the Expert Division, the floors in the rooms have different textures. Your robot will probably move slower on carpet than on linoleum. If your robot uses motions that are based purely on timing, you will need to test it with different power levels and floor surfaces.

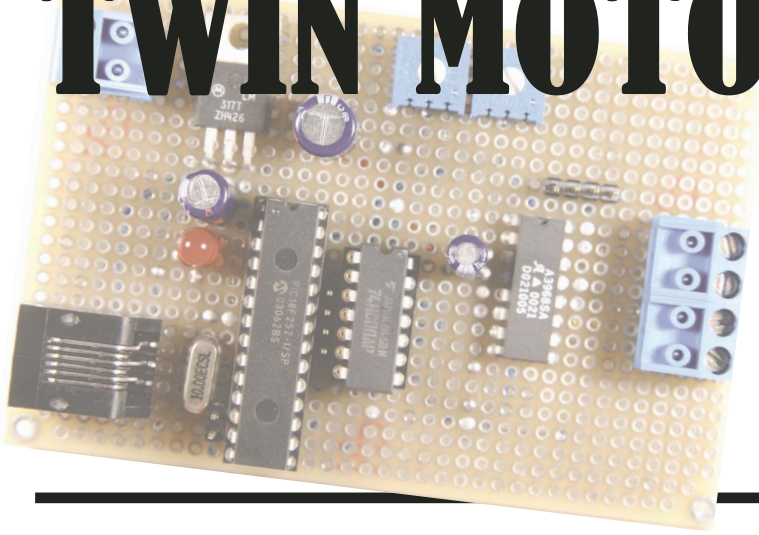
• *Keep it simple*. Many of the most successful robots rely on a simple design, simple software solution, and straightforward mechanics. A basic design that can navigate the maze, find the candle reliably, and put it out will always score well if it can succeed three tries in a row.

Expect changes to the environment. Almost all problems are followed by the entrant saying "that doesn't happen in my lab/school/basement." The most common problems are changes in lighting, including cameras, or other forms of interference. SV

FIREFIGHTING RULES

- Once turned on, the robot must be autonomous — in other words, it must be self-controlled without any human intervention.
- Robot must be able to fit in a box 31 cm long by 31 cm wide by 27 cm high. If the robot has feelers to sense an object or wall, the feelers will be counted as part of the robot's total dimensions.
- The robot must not use any destructive or dangerous methods to put out the candle. It may use such substances as water, air, CO_2, etc., but any method or material that is dangerous or will damage the arena is prohibited.
- It will be permissible to put out the candle by blowing air or other oxygen-bearing gas. However, this is not a practical method of extinguishing a fire in the real world. So, robots that do not use air streams to blow out the candle will receive a 15% time reduction.
- The maximum time limit for a robot to find the candle will be five minutes. After five minutes, the trial will be stopped. The maximum time for the robot to return to the Home Circle in the Return Trip mode will be two minutes.
- Any robot that slides along a wall will have an additional penalty point (one point equals one second) added to its time score for each 2 cm of wall it touches as it slides along.
- Any robot that touches the candle or its base with any part of its body or feeler, either deliberately or accidentally while the candle is lit, will have 50 penalty points (seconds) added to its Actual Time score each time the candle is hit.
- In order to make the contest realistic and to encourage the creation of smart robots, we have deliberately added uncertainty into the contest. The robot does not know in which of the four rooms the candle has been placed. Sometimes a robot gets lucky and the candle is in the first room it searches and sometimes the candle is in the fourth room searched.

Building a TWIN MOTOR DRIVER



Robots come in all sizes. So, you may not need a 100A motor driver to move your latest mechanical creation about. There will also be occasions when a couple of loosely coupled small motors are all you need to garner some motion from other articulated parts of your steel cowboy.

by Peter Best

To that end, for the last couple of weeks, I've been slinging wires between a PIC microcontroller and a dual full-bridge PWM motor driver that can put on the brakes. I would like to share my little twin-motor hardware design with you and at the same time show you how to code motion drivers for the Allegro A3968 Dual Full-Bridge PWM Motor Driver with Brake, on which my twin-motor driver is based.

While we're yaking and hacking, I'll also show you how to implement a motor speed control using the PIC's analog-to-digital converter in conjunction with the PIC's PWM module. As Larry of the famed Three Stooges would say, "Cut it out fellas. We've got work to do."

An A3968 Primer

The A3968 can be had in a pair of packages. To make this project easier to realize, I've decided to go with the A3968SA, which is the DIP version of the dual-motor driver. The A3968 is also available in an SOIC package and is specified as A3968SLB. If you really need a small dual-motor driver package, go with the SOIC version. Every nuance of the A3968 DIP design I'll present in this text applies directly to an identical and much smaller A3968 SOIC design.

The A3968 comprises a pair of H-bridges with each independent H-bridge having the capability of supporting a bidirectional ±650 mA motor load. The maximum motor voltage that can be applied to the A3968 is 30V. Motor winding current is controlled by an internal fixed-frequency PWM current control module. The use of this PWM-based current control circuitry is optional. We will use it in this design. Don't confuse the A3968's internal current control PWM with the speed control PWM that we will apply via the PIC's PWM engine.

The A3968's peak load current is determined by a combination of a reference voltage and a pair of current sensing resistors. The pulse duration of the A3968's internal fixed-frequency PWM current control module is set using a simple external RC network. The capacitor in the RC network I just mentioned also enables a blanking function to prevent false triggering of the A3968's current control PWM during switching transitions.

A pair of inputs ($INPUT_A$ and $INPUT_B$) determines the load current polarity (fancy words for motor shaft direction) by selecting the proper pair of source and sink drivers within the H-bridge. The brakes are applied when both of the $INPUT_A$ and $INPUT_B$ inputs are presented with logic low levels. Braking is the result of both of the selected H-bridge's source drivers being turned off and both of the H-bridge's sink drivers being turned on. All of an H-bridge's drivers can be disabled by applying a logic high level to both of the H-bridge inputs. Thermal and crossover-current protection is done in the A3968 hardware eliminating the need for any external protection circuitry.

Figure 1 is a representation of one of the H-bridges contained within the A3968. Note the two inputs and protection logic are integrated within the A3968's control logic circuitry. Before all is said and done, we will hang a resistor and capacitor (RT and CT in Figure 1) from the A3968's RC pin, complete each of the H-bridge sense circuits with a precision sense resistor mounted between the A3968's SENSE pin and ground, provide a voltage reference via a precision voltage divider to the A3968 REFERENCE pin, and add a load supply bypass capacitor across the H-bridge's VBB power input pin.

As you have already ascertained, the hardware assembly will be a piece of cake. Let's move on and determine the values of the very few components we will be parking around the pins of the A3968 DIP package.

A3968 Arithmetic

As I alluded to earlier, the A3968's internal current-control PWM circuitry controls the load current to each of the

motors that are attached to the A3968 H-bridge outputs. When the A3968 H-bridge outputs are turned on, current will begin to flow in an increasing manner through the motor winding. The load current is sensed by the H-bridge's Current-Sense Comparator by way of that precision current sense resistor we will be installing on the A3968 SENSE pin. Take another look at Figure 1. When a comparator is employed, you'll usually find a reference voltage somewhere in the mix. The reference voltage in this case is supplied by a pair of precision resistors that make up a precision voltage divider that taps in through the A3968 REFERENCE pin.

When the Current-Sense Comparator input voltage gleaned from the SENSE pin causes the Current-Sense Comparator to trip, the output of the Current-Sense Comparator resets the source-enable latch, which turns off the H-bridge source driver. At this point, the current will recirculate through the H-bridge sink driver and ground-clamp diode. As the load current recirculates, it decreases allowing the A3968's internal clock oscillator to override the Current-Sense Comparator's output and set the source-enable latch when the comparator's output falls below trip level. When the source-enable latch is set, the H-bridge source drivers again turn on, allowing load current to increase and the cycle to repeat.

The maximum reference voltage that can be applied to the A3968 REFERENCE pin is 2.0V. The load current trip voltage is related to the reference voltage and sense resistor value as follows:

$I_{TRIP} = V_{REF}/(4R_S)$

where:
I_{TRIP} = Desired Maximum Load Current
V_{REF} = Reference Voltage applied to A3968 REFERENCE Pin
R_S = Sense Resistor Value in Ohms

To be conservative, I'll keep the maximum load current value around 500 mA. The easy thing to do here is determine the sense resistor value that will give us 1A with the maximum reference voltage of 2.0V. Then we can simply cut the reference voltage in half, which should cut the ITRIP value in half. Performing the substitutions requires the sense resistor value to be 0.50Ω to end up with a maximum load current trip point of 1A with a reference voltage of 2.0V. The closest I could come to 0.50Ω in a half-watt SMT package was 0.51Ω, which is close enough. I know, I know. I said this is not an SMT project. However, I like to use SMT passives when I do point-to-point wiring projects as it saves space and time so I don't have to stop and cut leads every time I mount a diode, resistor, or capacitor.

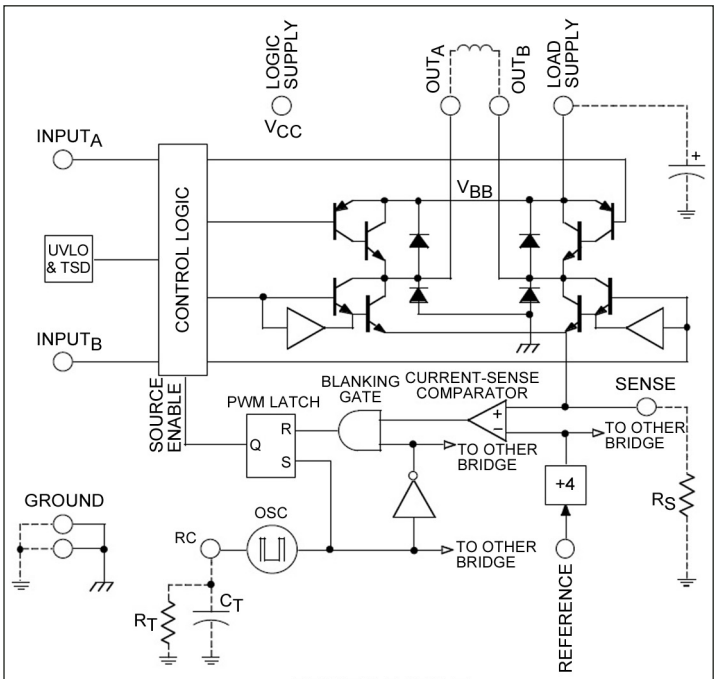


FIGURE 1. Note the absence of MOSFETs. The Allegro folks call these special transistors Satlingtons as they combine the low voltage drop of a saturated transistor and the high peak current capability of a Darlington.

TRUTH TABLE

$INPUT_A$	$INPUT_B$	OUT_A	OUT_B	Description
L	L	L	L	Brake mode
L	H	L	H	"Forward"
H	L	H	L	"Reverse"
H	H	Z	Z	Disable

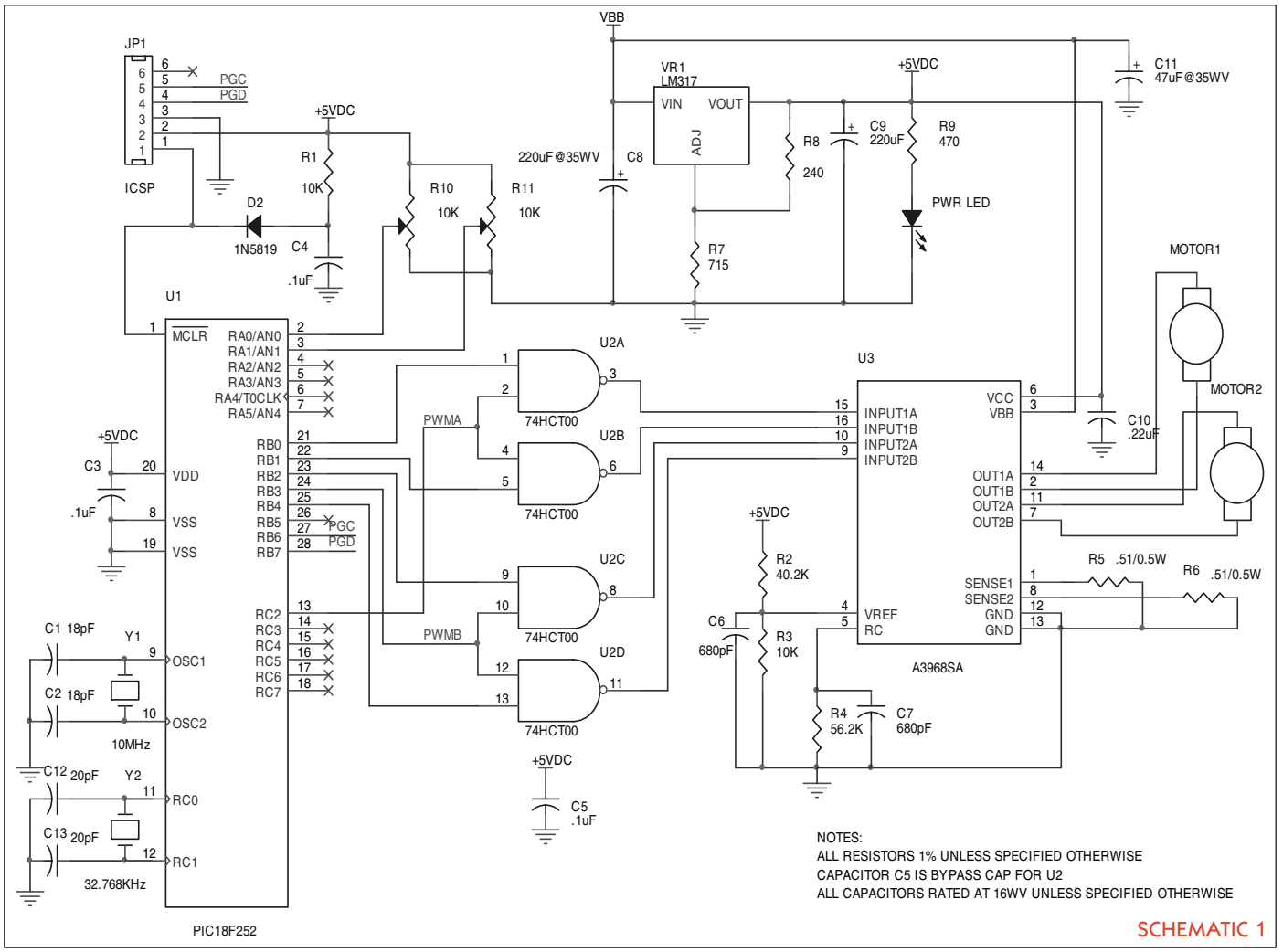
Z = High impedance

The next step is to put a reference voltage as close to 1.0V as possible at the A3968 REFERENCE pin. With an ideal source voltage of +5.0V, a 40K/10K voltage divider combination would be perfect. This is not a perfect world as 40K is not a standard 1% value and I doubt if I can generate exactly +5.0V from my regulated power supply circuit. So, the next best thing is a standard 40.2K 1% resistor and a 10K 1% resistor in my reference voltage precision voltage divider circuit plan.

Applying good old Ohm's Law gives a voltage drop of 0.996V across the 10K resistor, which is tapped into the A3968 REFERENCE pin. Jumping into the future, I measured my actual logic supply voltage at 4.97V, which theoretically yields 0.990V at the REFERENCE pin. I physically recorded 0.987V at the REFERENCE pin. Applying our mathematical known values to our I_{TRIP} formula results in a maximum current trip point value of 484 mA for the physically measured reference and power supply voltage values and 488 mA if we use the theoretical calculated reference and power supply voltage values.

The A3968 datasheet saves us some computational time with respect to the RC network values. RT and CT have datasheet recommended values of 56K and 680 pF, respectively. This combination of values results in a nominal oscillator frequency of 25.4 kHz.

The turning on of an H-bridge source driver produces a current spike that can trip the Current-Sense Comparator and reset the source-enable latch at the wrong time. To prevent this from happening, the Current-Sense Comparator output is blanked for a period of time when the source driver is activated. The blanking time is determined by the capacitor C_T. Without getting into the simple math involved, the 680 pF capacitor in our RC network blanks the comparator output for 1.3 µS. A 1.8 µS blanking



SCHEMATIC 1

period is automatically generated by the A3968 itself when the load current changes polarity. If you need more blanking detail, please check out the A3968 datasheet.

I'm sorry, but that's all of the math I can offer up right now. I guess we'll just have to build up some hardware now.

Three-DIP Wonder

You're already up to speed with the components that are supporting U3 in Schematic 1. So, let's talk about the logic woven by U1 and U2 that supports the A3968. If you've ever done a PIC project before, you can see that there's nothing remarkable about the PIC18F252 configuration. The standard Microchip ICSP programming/debugging configuration consisting of JP1, R1, C4, and D2 is tied to the PIC18F252's MCLR pin. I've tied in a couple of 10K potentiometers to feed two of the five possible analog-to-digital converter channels. The pots are optional and will act as physical speed controls for MOTOR1 and MOTOR2. You can also control the motor speed via I^{2}C or RS-232 as the interface pins for those communications interfaces are open.

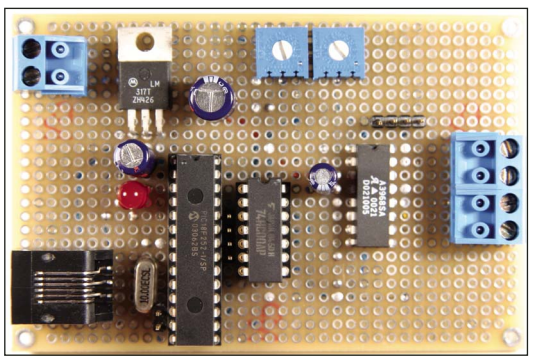


PHOTO 1. All you need is a Moto Tool, some wirewrap wire, and a soldering iron to assemble something similar to my A3968 driver board. I used the Moto Tool to ream out the holes to allow the screw terminal posts to feed through the perfboard.

The PIC is clocked at 10 MHz in this design. I chose 10 MHz for the base system clock because you can put the PIC into HSPLL (High Speed PLL) clock mode and run at 40 MHz, if you would like. Remember that the clock speed directly affects the PWM period and faster may not be better, depending on your choice of motors. The 32.768 kHz crystal drives the real-time clock, which I'll show you how to set up and use when we talk about the A3968 motor driver firmware.

All of the logic within U1, U2, and U3 is powered by a +5.0V regulated power supply. The resistor combination of R7 and R8 forces the output of the LM317 variable voltage regulator very close to the ideal +5.0V level. If you want precise voltage control, you may substitute a potentiometer for R7,

which will allow you to tweak the LM317 output. Capacitors C8 and C9 perform power supply noise filtering duty while filter capacitor C11 is dedicated to the VBB load supply input of the A3968. I chose the LM317 because it can easily handle the maximum 30V load supply voltage and still perform its +5.0V regulation duty.

To independently control the behavior of the motor pair requires both of the PIC's PWM modules. If you take another look at Figure 1, you'll see a truth table that defines the input sequences needed to move and brake the motors. The first rule we must make — which we will call Rule #1 — is to never allow the PWM to apply the brakes. To do that, we must make sure that one of the inputs is always in a logical high condition. Having both inputs go high at the same time is not a bad thing as it only disables the motors for that part of the PWM cycle. We never want the PWM input and the associated input to be low at the same time as that puts on the brakes between PWM cycles.

So, Rule #2 says that in the Forward and Reverse truth table entries, we will always apply the PWM to the input that is defined at a low level. Doing so will drive the motor when the PWM signal is low and disable the motor when the PWM signal is high. Now that we have determined how we must drive the motors using a PWM signal, let's examine the logic implemented by the 74HCT00 NAND gates.

Suppose we want to drive MOTOR1 forward. According to the truth table and our Rule #1, we must drive INPUTB high and apply the PWM to INPUTA. If you're not familiar with how a NAND gate works, it takes both NAND gate inputs to be high for the output of the NAND gate to go low. Otherwise, the output of the NAND gate is high. So, to force U2B to always show a high at its output, we feed its pin 5 with a low from PIC port pin RB1. If the pin 4 PWM signal at U2B swings high, the output at pin 6 of U2B will remain high, as a high on U2B pin 4 is NANDed with a U2B low on pin 5. When the PWM applied to U2B pin 4 swings low, both of the U2B inputs are low, which results in a high-level output on U2B pin 6. Thus, the output of U2B will remain high no matter what. That's what our truth table tells us we need to see.

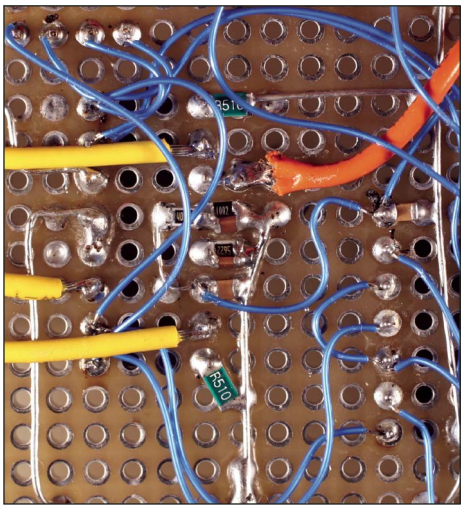


PHOTO 2. It may be ugly, but it works. You can easily identify the sense resistors in this shot. The precision voltage divider and the RC network are all stuffed into the center among the A3968 pins.

We need both input pins of U2A to go high to get a low on the U2A output pin. So, we will drive pin 1 of U2A high using RB0. When the PWM signal driving pin 2 of U2A goes high, the output pin of U2A will go low, supplying power to MOTOR1. When the PWM swings low on U2A pin 2, the load current applied to MOTOR1 will be removed as the output of U2A will be in a high state. A look at the truth table shows that if both $INPUT_A$ and $INPUT_B$ go high, the load current to the motor is removed. When the PWM swings high, $INPUT_A$ goes low while $INPUT_B$ remains high; this allows load current to pass through the motor winding.

To drive MOTOR1 in Reverse, we simply swap the logic levels at U2A pin 1 and U2B pin 5. The same forward/reverse motor drive logic I've just walked through applies to the U2C and U2D NAND gates driving the A3968's MOTOR2 inputs.

Applying the brakes to MOTOR1 is the act of driving the outputs of U2A and U2B low simultaneously. To do that, we must apply a logic high to both U2A inputs and both U2B inputs. This is accomplished by disabling PWMA and driving the RC2 pin high, while at the same time driving RB0 and RB1 high. Again, the same brake logic applies to U2C, U2D, and MOTOR2.

This is a good place to show you the actual hardware. Photo 1 shows my A3968 twin-motor driver layout. The load supply voltage enters via the removable two-position screw terminal in the upper left of the shot. The RJ-11 ICSP jack is mounted bottom left. Output to the motors is provided by the four-position removable screw terminal to the far right. The two sets of four pins you see in Photo 1 are tied to the inputs and outputs of the NAND gates. You don't see any glue components other than the filter caps in this shot as all of the passive circuitry is SMT and is mounted on the other side of the perfboard.

The 0805 SMT components fit well between the .1-inch-center plated-through holes of my custom plated-through hole perfboard. What you see in Photo 2 are the passive components that make up the precision voltage divider and the RC network used by the A3968. As you can see in Photo 2, I used standard wirewrap wire and point-to-point solder techniques to assemble my version of the A3968 driver board.

A Logical View of the A3968 Driver

I'm going to use a very nice tool called the Cleverscope to help me show you the relationship between the A3968 twin-motor driver firmware and the A3968 twin-motor driver hardware. The Cleverscope is a high-quality oscilloscope and function generator that interfaces via USB to a standard

PHOTO 3. This is a look at my Cleverscope's front panel. The logic level probes you see in Photo 4 use a flat cable to mate with the RJ-45 digital input jacks on the left.

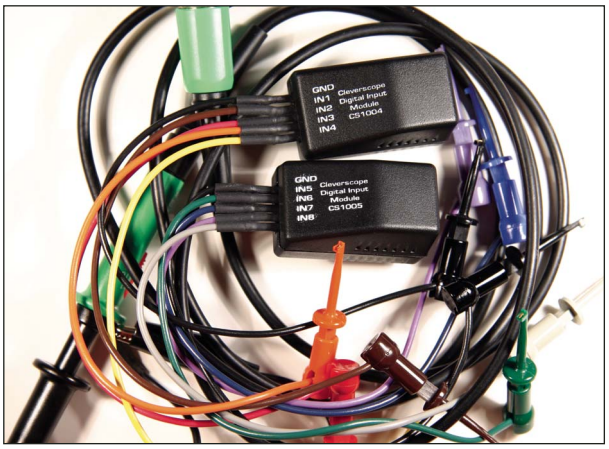


PHOTO 4. The Cleverscope comes standard with a pair of high-quality scope probes and a couple of the four-input digital input modules you see in the shot.

personal computer. The front panel of my Cleverscope is shown in Photo 3. A pair of very nice compensated scope probes comes as standard equipment with the Cleverscope. In addition, two digital input modules capable of monitoring four digital inputs each are also included with the Cleverscope package. (Now you know why I put those eight pins on my A3968 twin-motor driver board.) A representation of the Cleverscope scope probes and digital input modules can be seen in Photo 4.

I described how the logic for the A3968 flows through the PIC and NAND gates. Photo 5 is a Cleverscope screen capture. I've posted the Cleverscope Control Panel and the Signal Information windows out front so you can see them and get a feel for what the Cleverscope can do. For instance, the Signal Information window tells us that the period of the PWM signals measured by the standard scope probes on the Cleverscope's Channel A and Channel B is 1.64 mS. It also informs us that the pulse length is 819 µS and the duty cycle is 50%. If you are interested in voltages pertaining to the pulse train, those values can also be gleaned from the Signal Information window. The top scope trace (Channel A) is the raw PWMA signal coming into the NAND gates from the PIC for MOTOR1. The bottom trace (Channel B) is the raw PWMB signal used by MOTOR2. I attached the Cleverscope's eight data probes to the inputs and outputs of the NAND gates. Note that the Cleverscope Control Panel allows me to give each of the probes a meaningful name in the upper left corner of Photo 5. Let's examine the digital logic levels and see if we can figure out what the motors are doing.

Let's start at the top. RB0 is low and RB1 is high, which means INPUT1A should see a high logic level from the output of U2A. Sure enough, if you look at the INPUT1A digital capture, it is showing a "1" for INPUT1A. Recall that Rule #2 says that we always toggle the truth table's low input with the PWM signal. In this case, that would be INPUT1B, which is toggling in step with the scope's PWMA waveform. According to the truth table, with INPUT1A held at a high logic level and INPUT1B toggling, MOTOR1 is rotating in the reverse direction. A quick look at INPUT2A and INPUT2B reveals that INPUT2B is held high while INPUT2A is toggled by PWMB. Another look at the truth table and we see that MOTOR2 is spinning in the forward direction. Another clue that MOTOR2 is spinning opposite of MOTOR1 is the opposite NAND gate arrangement of the logic levels of RB0/RB1 versus RB2/RB4.

Photo 6 is a Cleverscope view of what happens when we slam on the brakes to stop MOTOR2. The truth table tells us to apply a logic low level to both of the A3968 inputs to kick off the braking action in the

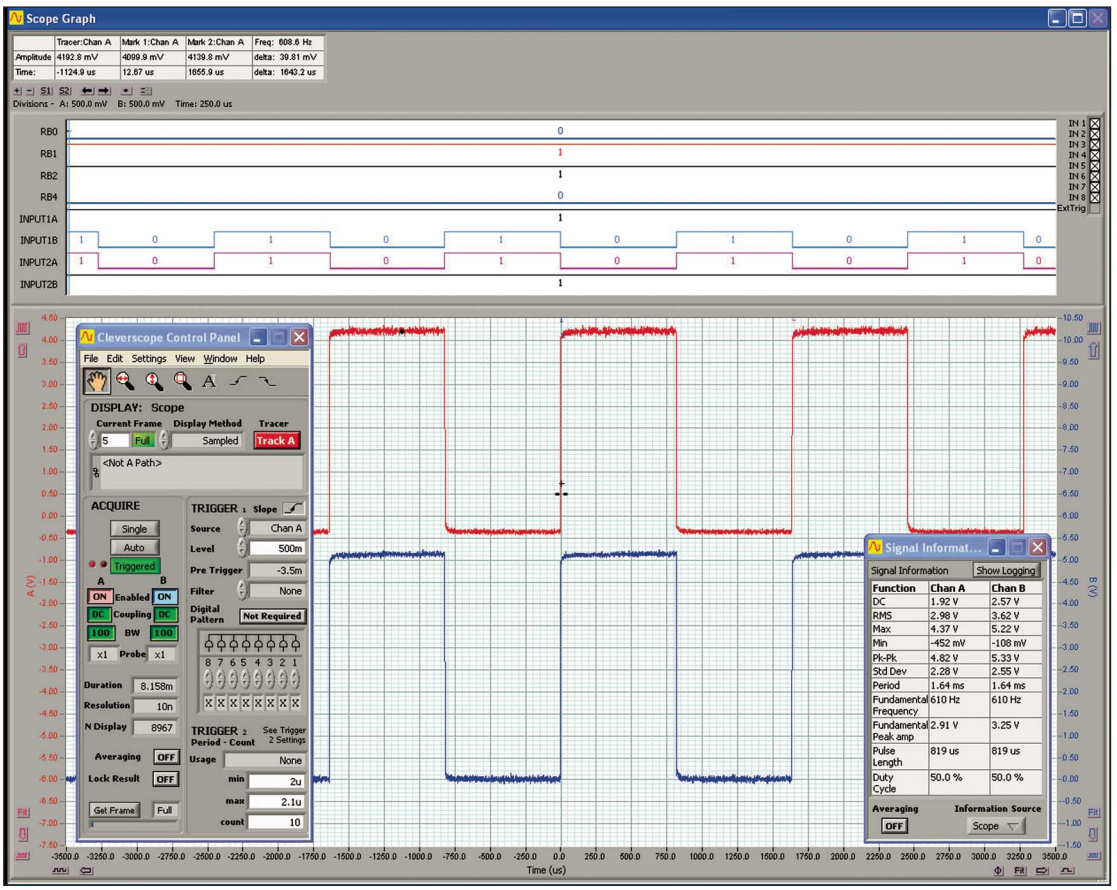


PHOTO 5. The Cleverscope is a wonderful tool that allows me to show you waveforms and digital logic levels on the printed page. I am very impressed by the Cleverscope's ease of use and accuracy. I have to beg my Tek scope to give me data that Cleverscope just lays out there for all to see.

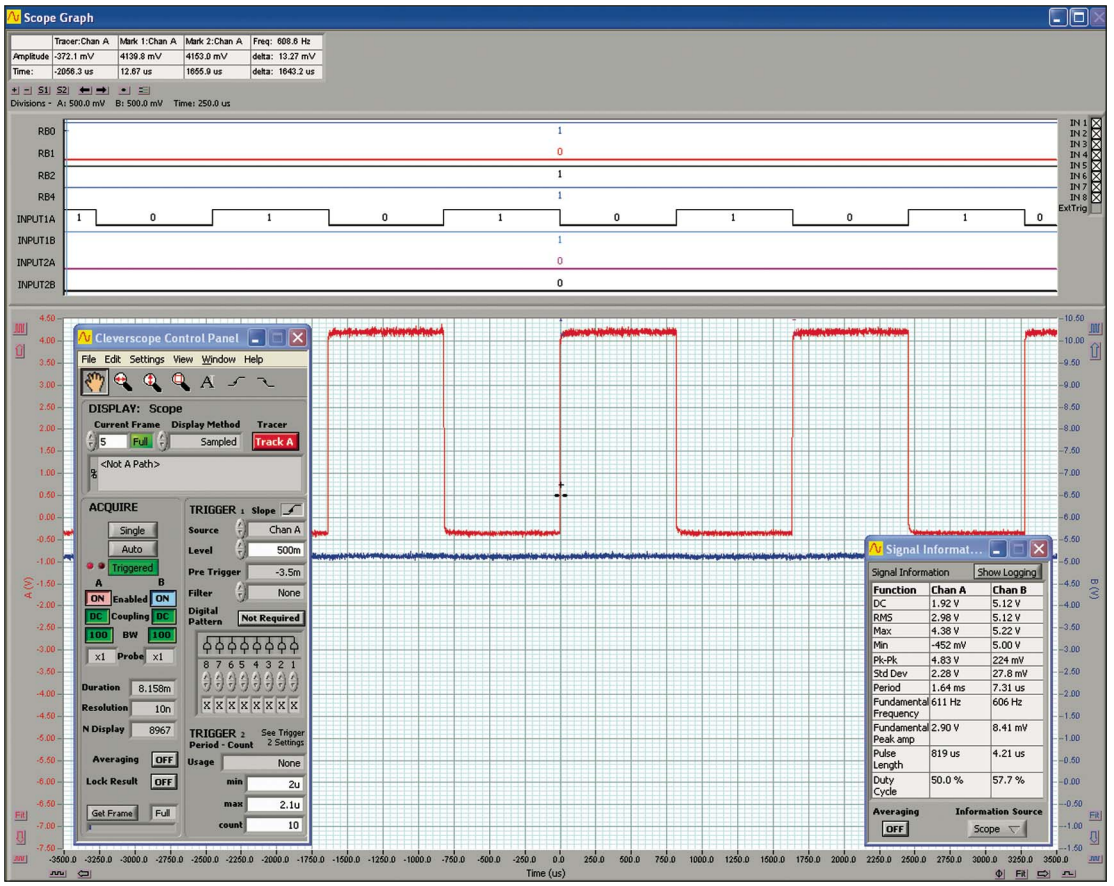


PHOTO 6. This Cleverscope screen shot shows that MOTOR2 has no PWM and all of the A3968 inputs that control MOTOR2 are held in a logic low state. MOTOR2 is not spinning the shaft at this point. However, MOTOR1 is humming right along.

H-bridge. Since we're muxing our PWM signals with NAND gates, we must pump a high logic level to all of the NAND gates involved with MOTOR2 to get low logic levels on the A3968 inputs. As you can see in Photo 6, RB2 and RB4 are at a logic level of "1" and the PWMB signal has been disabled to MOTOR2 and a high logic level is present on what was the MOTOR2 PWMB pin. The result is shown in the digital capture as INPUT2A and INPUT2B are both showing low logic levels. Trust me. The motor stopped on a dime. I'll bet you can figure out what MOTOR1 is doing by the signal levels presented to you in the Cleverscope MOTOR2 brake shot. Pretty "clever," huh?

Well, the driver firmware is just as clever. I've laid out all of the code necessary to move and brake a motor shaft in the Motor Functions area of Listing 1 (available on the *SERVO* website at **www.servomagazine.com**). The code is very easy to follow and directly associates with the forward/reverse/brake motor control logic flow I described earlier. So, I won't go into detail as to how it works. You can always send me a note with any questions you may have; peter@nerdvilla.com. My original code was compiled using the HI-TECH PICC-18 C compiler. I'm sure you will be able to convert my C source to your favorite language quite easily.

Timed Out

In addition to the A3968 twin-motor driver firmware, I've provided some very simple real-time clock driver code in the package. Basically, TIMER1 is configured to overflow once per second and generate an interrupt. My little real-time clock driver simply counts the overflows and converts the accumulation of TIMER1 interrupts to seconds, minutes, hours, and days. I show you how to use the forever-incrementing secs_timer variable in the APPLICATION AREA of Listing 1.

If you include the speed function in your application loop, you will be able to change the speed of each motor independently by adjusting the motor's associated potentiometer. The speed function code kicks off an analog-to-digital conversion that reads the voltage at the wiper of the selected pot. The 10-bit voltage value is then placed in the duty cycle register of the selected PWM module. Remember that the A3968 drives the motors on the low side of the PWM signal. So, your duty cycle will be based on the low-going PWM pulse time.

All of the parts for the A3968 motor driver are available from many of the vendors in this magazine. To make things easy, I'll offer a complete kit of parts for the A3968 driver on the NERDVILLA website at **www.nerdvilla.com**. And, don't worry. I won't make you type in all of the A3968 twin-motor driver source. The source code is available from the *SERVO* website (**www.servomagazine.com**) and **www.nerdvilla.com**.

I've got a lot more motor control gadgets I want to show you. However, I've motored right through my page allocation. So, until next time, have fun with your A3968 twin-motor driver board. SV

SOURCES

- A kit of parts for the A3968 motor driver board is available from **www.nerdvilla.com**.
- The Cleverscope is available from Saelig at **www.saelig.com**.
- The HI-TECH PICC-18 C compiler can be purchased from HI-TECH Software at **www.htsoft.com**.

The monthly magazine for the next generation of robotics experimenters

Seasons Geekings

Give your fellow Mech-Head a subscription to SERVO Magazine

Gift Subscription

Gift From:

Print Name _____

Address _____

City _____

State _____ Zip _____

Phone _____

Email _____

Gift To: ❑ 1 Year ❑ 2 Year ❑ 3 Year

Print Name _____

Address _____

City _____

State _____ Zip _____

Phone _____

Email _____

Gift To: ❑ 1 Year ❑ 2 Year ❑ 3 Year

Print Name _____

Address _____

City _____

State _____ Zip _____

Phone _____

Email _____

U.S. Prices:

1 Yr **$24.95**

2 Yr **$45.95**

3 Yr **$65.95**

To order, photocopy this page and mail to:

SERVO Magazine

Po Box 15277

North Hollywood, CA 91615

❑ Payment Enclosed
U.S. dollars drawn on a U.S. bank only

❑ Bill Me *(U.S. Orders Only!)*

❑ Visa/MC/Amex/DiscoverCard

Card # _____ Exp. Date _____

Name on card _____

For faster service, subscribe by phone or online

877-525-2539 (toll free)
818-487-4545 (outside US)
www.servomagazine.com

When ordering, use promotion code Y6FXMS

Beginner's Robotics on $50 a Month

– by Paul Pawelski

Amateur robotics is FUN.

Amateur robotics is EDUCATIONAL.

Amateur robotics is EXPENSIVE!

The sad fact of the matter is that the two easiest ways for a beginner to get into robotics are both very expensive. The most well-supported tools for the beginning roboticist are the LEGO Mindstorms and Parallax BASIC Stamp BoeBot. Both of them are used extensively for teaching robotics in schools and have literally thousands of websites devoted to using them. In addition, nearly any robotics club will have one or more persons in it that has experience with these systems. You will be able to get plenty of help if you start with either of these systems.

Unfortunately, either a Mindstorms set or a complete BoeBot kit will cost just under $200. This is a significant amount of money to come up with at one time for many people. Whether you are a teenager on an allowance or an adult trying to make house payments, car payments, student loan payments, etc., it is hard to allocate that amount of money at one time.

Then there is the problem of your dream bot. The bot you plan to build "someday." Whether it is a mini grand challenge racer to tear around the local BMX track or a 20-degree-of-freedom biped walker, there is something in the back of your mind that is your goal. Odds are that it is complex enough that you will need to use something other than a BASIC Stamp or LEGO RCX microcontroller to control it. At some point, you will have to learn how to use another microcontroller.

You could start from the beginning

with a PIC or AVR microcontroller and build a bot from scratch. Many people have and a lot of them have posted details of what they did on the web including parts lists. You could base your work off of their designs. This has the advantage of allowing you to buy your parts a few at a time to keep the monthly cost low, even if the cost for the whole project is more than you can afford at one time. However, you then have to face looking at a slowly growing mound of parts for months instead of a working robot. It is hard to keep motivated when you don't have something up and running.

The goal of this article series and its related online material (**www.servomagazine.com**) is to get you started in robotics for $50 a month — about the cost of a new video game. During the first month, you will buy the tools and supplies that every roboticist should have and you will learn how to solder (don't worry, it is easy once you see the trick). I will also show you some of the tricks to buying parts that will save you money. Next month, you will build a complete, easily programmable, easily upgradeable robot. It will move, sense the environment, and react to what it senses. In the third installment, we will add IR, a LCD display, and bump sensors. In the fourth month, we'll add a second microcontroller, learn about microcontroller-to-microcontroller communications, and add servo positioned sonar.

Buying on the Net and in the Stores

Even if you are lucky enough to live somewhere like Dallas, TX or Seattle, WA where "electronics supply store" means something more than the back third of RadioShack, you will probably end up buying a lot of your components over the Internet. Remember two things: Google is your friend, shipping costs and vague descriptions are you enemies.

Online stores range from such well-known places as Jameco, to guys who made bulk purchases and are trying to sell off the excess. Google helps you find them all. A problem with many of the smaller sites is vague descriptions. A site might list a 1,000 µF capacitor. Is it axial or radial? Mylar or electrolytic? 100V or 16V? It is only a good deal if you get what you need. If the price is really good, you might try emailing for more information; otherwise, look for another supplier.

Comparison shopping on the net is made more difficult by the dreaded shipping and handling monster. Some companies charge S/H based on weight, some charge based on total component cost, some by distance, and a select few even offer a flat rate. Except for the flat rate and part cost based shippers, you will never know the actual cost of an item until you are most of the way through checkout. This is why I tend to use sites that have a fixed shipping charge.

There are times when you will get a better deal from a company with variable shipping charges. For example, Circuit Specialist (**www.web-tronics.com/webtronics/index.html**) sells LEDs in bags of 100 for $1.50 and bags of 1,000 same value resistors for $3.00. I use them when I am getting supplies for a soldering class because the per-component savings adds up to more than the extra shipping cost.

The point is, you will have to take your time at first and research many suppliers. After a while, you'll get a feel for who has the best deals and you will develop a list of favorite places you go to first. Three of the companies at the top of my favorites list are All Electronics (**www.allelectronics.com**), Solarbotics (**www.solarbotics.com**), and The Mark III Robot Store (**www.junun.org/MarkIII/Store.jsp**). All Electronics has flat rate shipping. Solarbotics has flat rate for anything they can fit in a padded envelope. Mark III bases shipping on the total component cost, but has the same cost for any purchase up to $49.99.

There are occasions when saving time is more important than saving money. At 10:00 AM on a Saturday, you may discover that you can't do anything that you had planned for the weekend because you ordered 10Ω resistors instead of 10K ohm resistors from your favorite Internet supplier. In this case, you'll be happy that there is a RadioShack, Fry's, or some other electronics supply store within driving distance. Get to know the local suppliers. Even with the recent closure of many stores, RadioShacks are easy to find in the US, but there may also be other suppliers in your community. Stop in at a local TV repair shop and ask them where they go for parts (after letting them know that you are not going to be working on TVs).

If you are really lucky, you will find stores in your area that sometimes even beat the cost you would pay on the net for an item. For example, the local Harbor Freight store had a sale recently where multimeters were only $4.

Often the best deal is to find someone who has put together a kit that does what you want. Wright Hobbies (**www.wrighthobbies.net**) has put together a kit of parts to go with each article in this series. You can hunt around for lower cost suppliers, but you will be hard pressed to get all the parts listed for a lower price. Since Wright Hobbies bought parts in bulk for 50 to 100 kits at a time, they got a much lower cost on parts than you will get buying just the parts you need to do the project. This allows Wright Hobbies to sell the complete kit at a lower cost and still make a profit. In addition, your shipping cost is less since there is only one package to be shipped.

The Tools Every Beginning Roboticist Should Have

There are certain tools and supplies that will come in handy no matter what type of robot you are building. The tools and parts that are provided in the kit that goes with this month's article are listed in the Parts List. These are all hand tools and common supplies that can be purchased from many sources. As a kit, the items cost just under $50 including shipping. Bought separately from multiple sources, the best deal I could manage for them was around $55 including shipping.

Many people will recommend that you look at tools as an investment and buy the best tool that you can afford. This makes sense if you are buying something like a milling machine. A

cheap tabletop milling machine can be purchased for $350 and will last maybe 10 years in light use. A high-quality milling machine will cost at least $1,100, but will last 40 years and make more accurate cuts than the cheaper machine. In terms of dollars per year, the high-quality milling machine is the better deal since you will only need to buy one. However, the items on this list are so useful that most people who have been into amateur robotics for a long time have multiple copies of each item. For now, concentrate on getting serviceable items. Later, if you stay in the hobby, you can get higher quality versions for your workbench and these serviceable tools will become the ones you take with you when you go to competitions or club build nights. Remember, the more you spend on tools, the less you will have for robot parts!

The most useful tool you can have is a multimeter. No matter how careful you are when building a bot, you will make mistakes. You will put the batteries in and it won't run, or it will just turn left, or any of a dozen other surprises will happen. The multimeter will be the first tool you will reach for to debug the problem. Using the ohmmeter function of a multimeter, you can easily check each trace to look for loose connections or bad solder joints and see whether the sensor you hooked up just fried. Using the voltmeter function, you can make sure your components are getting the right voltage, the I/O port is actually sending a signal, and the batteries have some juice left.

This isn't precision work and any digital multimeter will do. The Harbor Freight meter mentioned previously will meet your needs for now. Even when it is not on sale, Harbor Freight sells them for $10 and All Electronics has them for $7. The only thing that this low-cost meter doesn't have that would be useful to the beginning roboticist is an audible continuity tester. The continuity tester produces a tone when there is a low resistance path between the test probes. This allows you to check paths without having to look at the readout of the multimeter. The meter included in the kit for this month's article includes a continuity tester.

The next tool you need is a wire cutter/stripper. The $2.50 one listed in the parts table can be adjusted to strip various sized wires using a stop screw. This is a bit of a pain if you are using wires of many sizes, but for the most part, you will be working with one size wire to begin with. Twenty-two gauge solid copper wire can easily handle two to three amps constant current without damage. It will fit into breadboard sockets and is fairly easy to solder.

Small pliers are a necessity as are regular and small screwdrivers. Flat nose, needle nose, and bent nose pliers can be bought for around $2.50 a piece. For the same price you can get various cutters — such as flat, chain, and angled — that come in handy for trimming components. A set of four flat nose and Philips screwdrivers only cost about $1 at WalMart and many other general merchandise stores. A set of precision (i.e., small) screwdrivers will cost about $2.50. Most stores will also carry electrical tape and small hobby knifes. Each should cost under $1.

Finally, you will need a soldering iron and its related supplies. There is a significant difference between the lowest cost irons and the mid-range irons, but with a little work you can get the cheap iron to work for you. It will be all right for light duty work. This will be the first item you will want to upgrade as you get into more complex projects. However, an $80 Weller variable temperature iron would break the $50 a month goal of this series, so we'll start with the cheap iron — 30- or 40-watt with a very narrow tip.

A cheap iron will not have thermal feedback control. Its temperature will drop quickly when you touch it to a component. This is not a bad thing for a beginner since it will get you into the habit of not loitering while making your solder joints. Spending too much time when soldering a joint is bad because it gives the heat time to transfer from the joint to the components you are trying to connect. Many electronic components are heat sensitive. An electrolytic capacitor or a microcontroller can easily get fried if you waste too much time soldering one of their leads.

Soldering requires more than a soldering iron. You want a holder for your iron so that you don't put the hot tip on your table. You also want some sort of holder for the items you are soldering and a sponge to keep the tip of the iron clean. The kit for this article from Wright Hobbies includes a nice little all-in-one holder for your iron, your work, and a sponge. You will also need some toothpicks (about $1), as well as solder and the beginning solderer's best friend — rosin paste.

PARTS LIST

DESCRIPTION	QTY.	EST. PRICE
• Multimeter	1	$9.99
• Wire Cutter/Stripper	1	$1.99
• Combination Pliers	1	$1.99
• Bent Nose Pliers	1	$1.99
• Closecut Diagonal Pliers	1	$1.99
• Sand Paper	1	$1.00
• Soldering Iron w/Helping Hands Holder	1	$9.95
• Flux Paste	1	$2.00
• Solder 60/40 0.03" dia.	1	$2.00
• 9V Battery Connector	1	$0.25
• 40 Pin Header	1	$0.50
• LEDs 10 count pkg.	1	$1.29
• 360 ohm Resistors, 10 count pkg.	1	$1.89
• Multipurpose PC Board	1	$1.49
• Screwdriver Set	1	$2.50
• Precision Screwdriver Set	1	$2.50
• Electrical Tape	1	$1.00
• Toothpicks	1	$1.00
TOTAL (before S/H)		**$45.32**

There are many different sizes and varieties of solder that you can buy. I won't go into the differences here, but I will recommend that thinner solder is better for electrical work, especially when using a cheap iron. Since a cheap iron will cool fast when it touches any item that either absorbs or conducts heat, melting a big, thick piece of solder is hard to do. In addition, beginners have a tendency to use too much solder. Using thinner solder will help avoid that tendency. A small tube of 60/40, 0.03" solder will cost under $2.

Now, you need some items to use while practicing soldering. You need to get a 10 count package of LEDs ($2.59), 10 360 ohm resistors ($2), a 9V battery holder with wire leads ($.89), and a multipurpose PC board ($1.61). All of these things are in the kit and are also available at RadioShack. You will also need a 40 pin header ($0.50) which is in the kit or can be ordered from many online electronics stores such as All Electronics.

Soldering

The online material that goes with this article can be found at **www.wrighthobbies.net/guides/**, as well as on the *SERVO* website at **www.servomagazine.com**. It contains instructions for a project that will teach you how to solder and how to troubleshoot your circuits if they do not work right the first time. The project consists of making multiple copies of a simple LED circuit. The instructions were placed online so that we could go into more detail than would be possible in the limited space available here.

Conclusion

By the time you have made five to 10 iterations of the circuit described in the online instructions, you should be fairly comfortable with soldering and bad joints should be rare. You now have the basic tools and the basic skill needed to build a robot!

Next month, we will build a complete robot. This robot will be more than just a platform. It will be mobile, sense its environment, change how it moves based on what it senses, and be easily modifiable.

The robot will be programmed using BASCOM-AVR from MCS Electronics. You can download a free version of BASCOM-AVR complete with manual from **www.mcselec.com/index.php?option=com_content&task=view&id=14&Itemid=41**. BASCOM runs on Windows 95 or newer PCs. The programmer we will be building uses the parallel (printer) port so you will need to have a PC available with a parallel port. If you do not already have one, you can often pick up an older PC at a computer repair shop or a second-hand store for less than $50. Just make sure that the computer's operating system is Windows 95, 98, 2000, NT, or XP. Until next month, have fun with your project! SV

You know it wants one. Go ahead, spoil your robot. Give it the brain and the brawn to take on other robots.

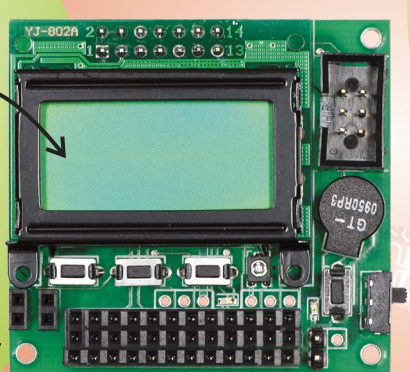
Give your robot the power to do anything it can imagine.

Baby Orangutan *starting at $24.95*

Baby Orangutan: great main controller for tiny robots or auxiliary controller for bigger bots. mega168 uC with 2-channel H-bridge.

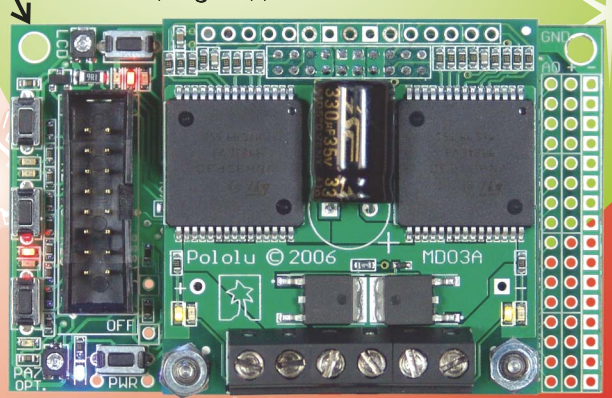
All modules shown actual size.

Orangutan X2: mega644 uC provides plenty of horsepower to match one horsepower of motor drive (peak). Integrated USB programmer. User buttons, potentiometer, buzzer, LCD or PLED display support, and much more.



Orangutan: full-featured robot controller perfect for small robots. mega168 uC, 2-channel H-bridge, 8x2 LCD, buzzer, and more.

Orangutan *starting at $49.95*



Orangutan X2 *starting at $99*

Pololu
Robotics & Electronics
6000 S. Eastern Ave. 12D, Las Vegas, NV 89119

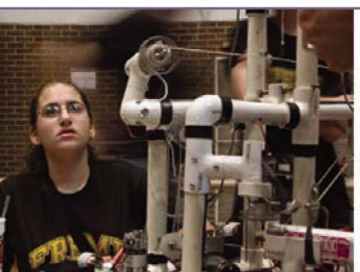
Find out more at **www.pololu.com** or by calling 1-877-7-POLOLU

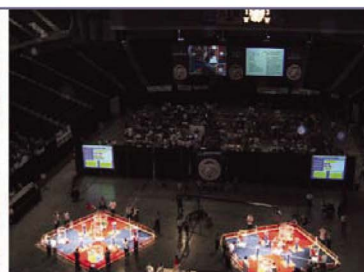
BEST
(Boosting Engineering, Science, and Technology)
Robotics Competition

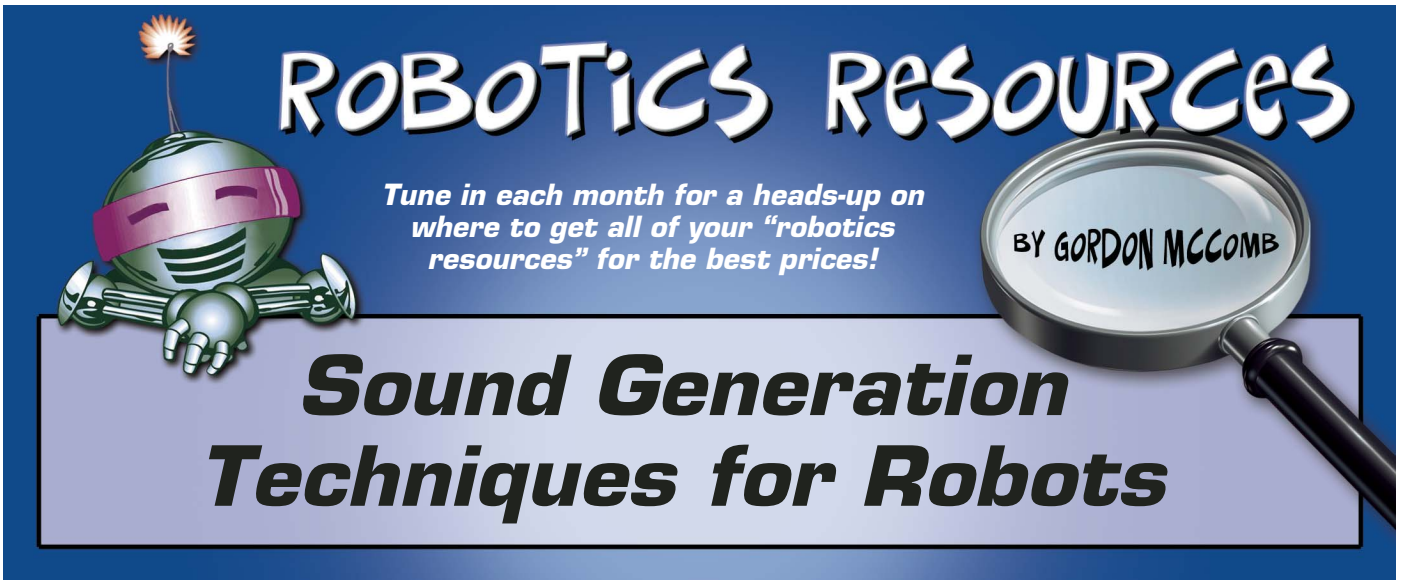
Come join the fun! Visit us at www.bestinc.org

SERVO MAGAZINE

is a proud sponsor of BEST







Of our five senses, we use sight and sound the most in order to survive. Often, what we can't see we can hear — a train coming around a corner, for instance. Though we may not be able to see the train's approach, we can hear its engine, and probably its warning horn. It's enough that we know to stay clear of the tracks.

We rely on visual and aural cues throughout our daily lives. Consider the average outing in the car: Stop signs change colors to tell us when to stop and when to hit the gas pedal. A crossing guard may use a whistle to get our attention. The flashing lights and wailing siren of an ambulance get us to pull over so they can pass and attend to the injured.

So, it should be the same with our robots. We can use cues of various types to help us understand the robot's current condition, or to notify us of some impending danger so we can intercede, and possibly prevent mechanical damage. LEDs and LCD displays are commonly used for visually annunciating a robot's status; so too can sound be used to help the robot communicate with us.

In this month's Robotics Resources, we'll take a look at affordable sound generation techniques for robots, from simple noise makers to sophisticated voice synthesizers.

Preprogrammed Sound Modules

At the bottom of the sound food chain is the preprogrammed or "canned" sound module, typical in such products as greeting cards or musical ornaments. Most are programmed with a song, though a few — like the electronic whoopee cushion — are meant to emit a sound effect.

Most sound modules are completely self-contained, including a small speaker (piezo or dynamic magnet), a button battery, and some means to set it off, usually a small pushbutton switch. You can salvage the sound module from a greeting card or other product and reuse it in your robot. Craft stores are a good source of new sound modules that can be added to homemade ornaments.

Controlling a sound module from your microcontroller or PC involves triggering the module by electrical signal, rather than by using its mechanical pushbutton. Depending on the design of the module, you may be able to trigger it simply by pulsing it with a short five-volt signal; with others, you may need to wire a small reed relay across the switch terminals, then activate the relay from your controller or computer. When operating external devices, it's always a good idea to use a buffer IC on any output to prevent damage to your controller or PC.

Siren and Other Warning Devices

Similar to the programmed sound module are siren or warning devices used with small personal security items. These are found in battery-operated alarms, such as the type that are hung on a doorknob in a motel room. If the doorknob is turned when the alarm is armed, it emits a shriek through its built-in piezo speaker.

These siren and warning modules are sometimes available surplus, but it's more common to just pull the whole sound-making module out of the alarm, and reuse it on a robot. Assuming you are needing your robot to act as a security guard, the main benefit of a siren module is the volume it produces. Ordinary programmed sound modules like those described above have fairly low volume levels — you need to be no more than a few feet away to hear it. With a siren module from a portable alarm, you can hear it from 100 to 300 feet away.

Connecting the alarm electronics to your robot depends on the design of the alarm. Some use a mechanical tilt or mercury switch, so the same interfacing requirements are encountered as with a preprogrammed sound module. Those that are triggered by an electric pulse — from an optical tilt switch, for instance — can usually be directly interfaced to a controller or PC without arcane means such as a relay (but still use a buffer IC in between).

Sound Effects Kits

Apart from the venerable blinky lights, sound effects kits are among the most popular, because when constructed, they "do something." Several companies manufacture and sell sound effects kits that you can use as self-

contained modules in your robot projects. For example, the light-sensitive Theremin Kit from Electronic Goldmine produces distinctly different sound effects by altering the amount of light falling on two sensors. The company also sells a 10 note sound kit and several others.

Most sound effects kits are designed to be self-contained. That means they come with an amplifier, if they need one, and a speaker. Controlling them on a robot requires interfacing the selector buttons to the outputs of your microcontroller or PC. Fortunately, most kits come with a schematic and you can more easily determine the voltage levels and current draw through the switch inputs.

Digital Sound Recording Modules

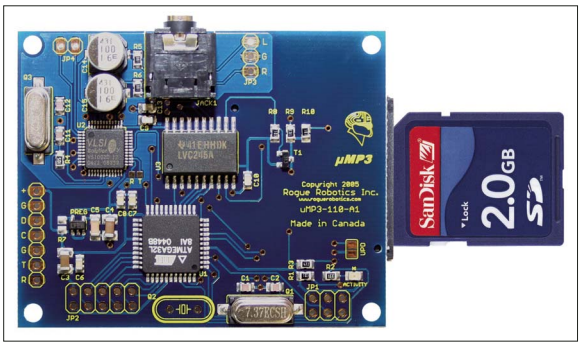
Thousands of projects have been built around the ISD ChipCorder series of sound recording modules (now manufactured and sold by Winbond). You can still purchase these chips from some electronics distributors, such as Jameco (**www.jameco.com**). You can also find them — or similar products — in ready-made products that you can salvage. I've found inexpensive digital sound recording and playback modules in everything from picture frames to talking flowers.

Recording time varies, but is usually five to 30 seconds. You can record your own voice or sound effect for later playback. The module has its own amplifier and speaker, or you can add one. As above, if the playback button is fully mechanical, you'll need to experiment with the best means of interfacing the device to your microcontroller or computer.

MP3 Recordable Sound Modules

Sound units built around the ChipCorder and similar products have a relatively limited recording/playback time. Sound quality is also somewhat limited to an 8 kHz monophonic range. That limits the devices to basic voice playback. MP3 modules let you store voice, sound effects, and even music on Flash memory, and play it back through an on-board MP3 decoder.

One way to add MP3 playback to your robot is to hack an inexpensive MP3 player. This is fine for a module that merely plays background music, but is insufficient if you're wanting the robot to play certain music, voice, or effects under program control. There are a number of MP3 development boards available that you can interface to your robot — depending on the power and capabilities of the microcontroller or computer operating your bot.



The Rogue Robotics uMP3 all-in-one MP3 player development board.

For example, the uMP3 from Rogue Robotics is a fully self-contained MP3 playback development board. It includes an onboard MP3 decoder chip (specifically the VS1011), low-power amplifier (you can use an external amplifier if you need more volume), and a Secure Digital connector for direct connection to an SD Flash card.

The product supports SD cards formatted with DOS FAT and FAT32 sectors, so you can prepare your music and other sound effects on your PC, then transfer them to the SD card for use with the uMP3. Playback is controlled via a serial or parallel port. In addition to MP3 files, the uMP3 will play files in WAV format, and will produce simple tones without the use of recorded files.

Along similar lines, but without all the features, is the MP3 breakout board from Spark Fun Electronics. It's mostly a breakout board for the surface mount MP3 decoder chip; if you want a completely self-contained sound-making apparatus for your robot, you'll need a means to send MP3 data to the chip. A common method is to read data from a Flash card, and send it to the decoder. Spark Fun sells a microSD Flash card breakout board, as well.

There's also the MP3 module from Olimex, which contains both the MP3 decoder chip, SD media holder, and headphone amplifier. Playback is intended to be controlled by a microcontroller; a 10-pin connector is provided to interface to a host. Or, the module can be used as a simple playback unit with special programming of the board's EEPROM.

Voice Synthesis Chips and Modules

The most elaborate type of sound making device is the voice synthesizer. These have been available in chip form on and off for the last couple of decades. The venerable General Instrument SP0256 speech chip is still sold by a few surplus outfits (see SpeechChips.com in the sources section), and for many years this product was about the only affordable alternative to quality speech generation.

New products have been introduced over the last few years, especially with the emergence of robotics as a bona fide hobbyist market. These include the Devantech SP03, Parallax EMIC, Magnevation SpeakJet, and the Savage Innovations Soundgin. The latter is a complex six voice sound synthesizer, in addition to being a voice chip.

All of these solutions are designed to be connected directly with a host controller or PC, either via a serial or a parallel port. Serial interfacing is perhaps the easiest as it requires just two or three wires. A parallel interface needs ground plus up to eight connections to provide a full byte-wide port. If your microcontroller or PC doesn't support the interface you need to work with the speech synthesizer of your choice, you can always use a parallel-to-serial or serial-to-parallel chip to convert one to the other. These chips are low cost, but they do add to the complexity of the overall design.

Most of the speech products — at least the chip-level ones — are not self-contained, meaning at the minimum you need to add an amplifier and speaker. And the chip-level speech synthesizers typically need a crystal, voltage regular, and additional components. The sellers of these products provide schematics for typical hookups. If you're interested in the GI SP0256, **Speech Chips.com** sells a handy circuit board that makes construction easy. They also sell the oddball 3.12 MHz crystal that the SP0256 design specs call for.

PC-based Sound Generation Software

Finally, if your robot uses a PC or laptop, you can turn to software-only solutions to generating sound. On the Windows side, for instance, are all sorts of sound editing and playback programs that you can make use of. Need to play the Star Trek theme when your robot enters the room? No problem! Get a recording of the song, and store it on the hard drive. To play back the song, load it into a sound program, and play it through the PC's sound card.

A few websites provide free or nearly free WAV recording/editing utilities and prerecorded sounds that you can use in your robot creations. For example, the Sound Effects Generator program from Windows Games lets you generate sounds by setting various controls. The full version of the program costs $10.

There's also a bevy of free recorded sounds, in MP3 file format, at **the recordist.com**. Most are short effects intended for things like games, but they are equally adaptable for robotics. The selection is from a larger library of effects. You can create your own sound effects from the many sound effects CDs that are for sale. Though copyrighted, you are generally free to use the sound effects for your own personal and non-commercial use. Check the license agreement that comes with the CD to be sure.

ABOUT THE AUTHOR

*Gordon McComb is the author of the best-selling Robot Builder's Bonanza, Robot Builder's Sourcebook, and Constructing Robot Bases, all from Tab/McGraw-Hill. In addition to writing books, he operates a small manufacturing company dedicated to low-cost amateur robotics (**www.budgetrobotics.com**). He can be reached at robots@robotoid.com.*

Sources

Arrick Robotics
www.robotics.com
Sells a speech synthesis module for the company's Trilobot robot, adaptable to most any robot design.

Carl's Electronics
www.electronickits.com
Do-it-yourself kits, including several sound effects kits.

Devantech Ltd.
www.robot-electronics.co.uk
Makers of the SP03 speech synthesizer module.

Electronic Goldmine
www.goldmine-elec-products.com
Sound effects kits including a light Theremin, 10 tone organ, and chirp-chirp kit.

Electronix Express
www.elexp.com
Variety of kits, including the Sound Effects Generator (includes speaker).

FlexiMusic — FlexiMusic Generator
www.fleximusic.com/generator/overview.htm
Windows-based utility for creating musical and non-musical sound effects.

Olimex
www.olimex.com/dev/mod-mp3.html
Makers of an MP3 development board based on the VLSI VS1000 series of MP3 integrated decoders.

Parallax, Inc.
www.parallax.com
Sellers of the EMIC speech module. Available in male and female voice versions.

Quasar Electronics
www.quasarelectronics.com
Do-it-yourself kits. See their section of audio and sound effects kits.

The Recordist
www.therecordist.com/pages/downloads.html
Free sound effects for games (and robots!).

Rogue Robotics
www.roguerobotics.com
Makers and sellers of the uMP3 self-contained MP3 development board.

Spark Fun Electronics
www.sparkfun.com
Developers and sellers of several unique sound-based products ideal for robotics, including an MP3 breakout board and a Flash media card adapter (holds data for the MP3 decoder chip).

SpeakJet
www.speakjet.com
Main home for the Magnevation SpeakJet speech processing IC. Available through dealers, which are listed on the website (includes Acroname, SuperDroid, HVW Tech, and others).

SpeechChips
www.speechchips.com
Sellers of a variety of speech-related products, including surplus General Instrument SP0256 speech chips, construction board for the SP0256, and the SpeakJet.

Soundgin
www.soundgin.com
Main home of the Savage Innovations Soundgin complex sound generator and speech synthesizer. Sold through dealers. Check the website for details.

Windows Games
www.windowsgames.co.uk/effects.html
Example of freeware/shareware sound generation software for Windows. **SV**

It has been slowly happening for the past couple of years. They were disappearing, one by one, until they can hardly be found any more. You were minding your own business, working contentedly on your projects, and didn't notice. Recently, you purchased a new computer and were shocked to find that it lacked a serial port. How could this happen? Your trusty friend the serial port, that let you communicate with all of your projects, had disappeared.

Serial ports have slowly been disappearing over the past few years and are being replaced by the USB port. In many ways, this is great because it makes people's experience using their computers nicer. For a hobbyist though, it can be a nightmare ... or at least it was in the past. This month's column will take a look at a couple different ways to transfer data to and from your robotic project.

Options for Data Transfer

The first way is to use a store-bought USB-to-serial converter. You could go to Office Depot or Best Buy and find one that spits out true RS-232 data, but there are a few problems with that. You are going to spend around $40 for the device, it will be in a big plastic shell, and you'll have to have a level shifter on your robot to convert the +10V to -10V signal into something that is actually useable. Luckily, you can avoid those troubles by using modules made by Parallax or Sparkfun. Both are great companies that genuinely support the robotics and electronic hobbyist community. Both Sparkfun and Parallax sell a module based on the FTDI FT232R chip. Sparkfun also sells a module based on the CP2102 chip by Silicon Laboratories. Parallax and Sparkfun sell the chips that they are using on their modules individually, so if you develop something that needs to be smaller or mass-produced, you can just buy the chips by themselves.

Available Features

These two chips are largely the same in the features that they offer. Both of them have all of the I/O that a standard serial port has. Almost no one uses anything but RX and TX anymore, but there are pins for RTS (Ready To Send), CTS (Clear To Send), DSR (Data Set Ready), DTR (Data Terminal Ready), RI (Ring Indicator), and DCD (Data Carrier Detect). These pins helped to keep buffers from overflowing back in the days when computers did their thing at very low speeds. Although it was not verified for this column, these pins may be useful again because these chips are capable of speeds much higher than your computer's serial port. The FTDI chip is capable of up to three megabaud! That is 26 times as fast as your old serial port could manage. The Silicon Laboratories chip tops out at one megabaud. That is still plenty fast for nearly anything that you would need to do with your robotic projects.

Both of these chips can run under either USB power or external power. Internally, they run at 3.3 volts and can supply some power to external

	CP2102	FT232R
Price for Quantity 1	$3.62	$4.10
USB Version	2.0	2.0
Clock	Internal	Internal – can be brought out at 6, 12, 24, or 48 MHz
Baud Rates	Standard rates up to one megabit	Standard and nonstandard rates up to three megabits
Buffer Sizes	567 byte RX, 640 byte TX	256 byte RX, 128 byte TX
Package	MLP28	28SSOP or QFN-32
Voltage Output	3.3V at up to 100 mA	3.3V at up to 50 mA
Drivers Available For	Mac, Windows, Linux	Mac, Windows, Linux
Internal EEPROM	1,024 bytes	128 bytes
Other		Can also use four of its pins as general-purpose I/O or eight of them if you don't use it as a serial port

Table 1. A comparison of the FT232R and CP2102 chips.

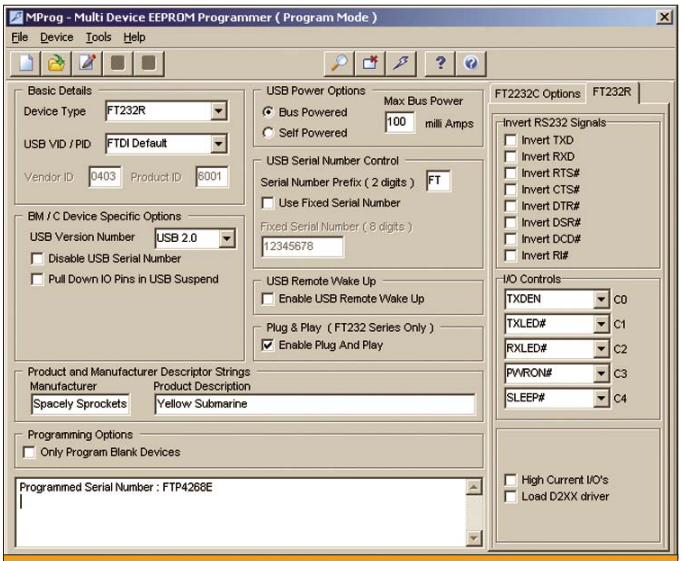


Figure 1. The MProg configuration utility.

circuitry. Both have an internal clock, but the FTDI chip lets you bring it out to drive your processor with. That is pretty handy. There are a bunch of other minor differences between these two chips. Table 1 shows a comparison of the capabilities.

Custom Options and Comparisons

Let's look at the customizations that you can perform with the FT232R and CP2102 chips. The most obvious thing that you can do with both is that you can change some of the data that the chip sends to the host computer when it 'enumerates.' Enumeration is a process that happens when you plug an USB device into a computer. During enumeration, the USB device tells the computer what type of device it is, various things like its name and serial number, and maybe most importantly, how much power it will draw at its maximum.

For the FT232R chip, there is a utility called MProg that allows you to change and upload these settings into the chip. MProg is a little quirky but does the job for you. The way that you can adjust the settings that are in a chip is to first go to the File menu and select New. This will give you a blank slate to work with. Since FTDI has several chips that this program works with, the first thing that you will need to do is select the type of chip that you are going to program. Now adjust all of the other settings to match what you want. The product description is what will show up in a little balloon on your desktop when you connect the chip to your computer for the first time. It is cool to see the name of your robot or other device that you may be working on show up when you connect it. (The name you use doesn't matter. The computer will still see the chip as a USB-to-serial.)

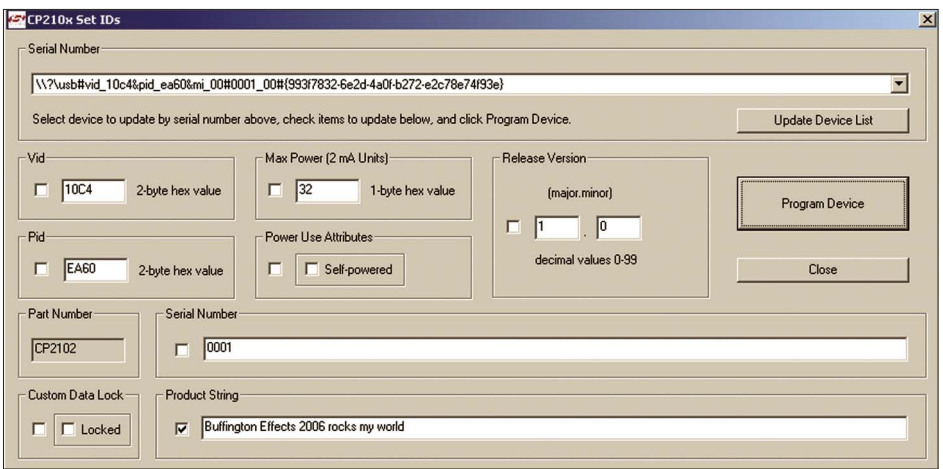


Figure 2. The interface for CP210xSetIDs.

The most important thing to set in MProg is the amount of power your device will draw from the USB port. The USB spec allows up to 500 milliamps to be drawn from the USB port. This current can be split up among any number of devices. The computer will keep track of how much power each device draws and if a device attempts to connect that would draw too much power, the computer will refuse to connect to it.

The FT232R chip has the ability to signal external devices when it is okay to start drawing larger amounts of power. When a device is connected, it is allowed to draw a maximum of 100 milliamps. Once it has successfully enumerated, it can draw up to the amount that it reported to the host computer.

The FT232R chip has five pins that can be used to signal various events. They can be used to drive LEDs to show that a receive or transmit is happening or they can signal that they can draw their full amount of power. You can get a 6, 12, 24, or 48 MHz clock signal from any of these pins.

Finally, you can set up this chip to be able to use four of these lines as I/O lines if you use a special driver. You will still be able to do serial communication through the chip at the same time. Alternately, you can set the chip into a mode where it stops being a serial chip and instead drives the serial pins as an eight-bit I/O port. This also requires a special driver.

As you can see, the FTDI chip is pretty cool. One drawback — at least for the hobbyist — is that the chip is pretty tiny. Unless you are fairly advanced, you will probably want to use one of the modules sold by Parallax or Sparkfun.

Let's take a look at the CP2102 chip's configuration.

Silicon Laboratories, which produces the CP2102, has a program called CP210xSetIDs.exe which lets you set up various things about the chip such as how much power it will be drawing from the host computer, serial number, and its product string. This program is very straightforward and easy to use. Simply plug in your device,

hit the Update Device List button, select your chip, change the settings, and hit Program Device. Silicon Laboratories gives away a library of functions that you can call from your own software that will let you set all of the things that you can enter in CP210xSetIDs. One notable difference between this chip and the FT232R is that when you connect this chip, your computer won't play the USB connect tones and won't show the little balloon that tells you what it is that was connected. Your device manager will see the device, though.

For this column, the drivers that were available on Sparkfun's website were used. If you want your device manager to show the device as something other than "SFE USB to RS232 Controller," you will need to compile your own drivers. This isn't as bad as it sounds. Silicon Laboratories provides code that you can use to do just that.

You might be wondering which chip is better. If this column were to make a recommendation, it would be to use the FT232R for everything except extremely cost-sensitive applications. It is nice that this chip causes Windows computers to announce that they have a new USB device connected. You can rename the USB device to be whatever you want without having to recompile the drivers, and finally, you can achieve triple the speed of the CP2102. Both are good chips, however. If one is out of stock, there isn't any reason that the other wouldn't work just as well for your robot, unless you need more than one megabaud.

Test Time

Once you have your drivers installed, you will probably be itching to test these chips. A quick way to test them is to simply put a jumper between the RX and TX pins. Use a terminal program to send data out of the serial port. Windows comes with a program called HyperTerminal. This program works, but is a pain to use since it expects you to use it with a modem instead of just connecting to the serial port directly. A good program to use is called serialterm. You can find this program online for free. It runs in a command prompt so it definitely isn't fancy, but it does a good job for simple serial communications. To use serialterm, just type from within a command prompt:

serialterm <com port> <port speed>
ex: serialterm com4 9600

Where To?

One thing that hasn't been covered is how do you know where your USB port showed up? If you are running Windows, you can go into your Device Manager and take a look at the 'Ports (COM & LPT)' section. Your serial port will show up there. For example, it may look like 'SFE USB to RS232 Controller (COM4). In that example, you would know that you need to communicate with com 4 when you want to communicate with devices that are connected to your converter chip.

Wrap-up

Being able to communicate with your robot using a serial port is a very handy thing. It can sometimes be a difficult thing to figure out what is going on with a robot when it is doing higher level processing. Having it spit out short strings with a variable value or that tell you what routine it is currently running can be a huge help. It is also handy to be able to chart data over time. By sending a number or two every once in a while to a PC, you can take that data and display it in programs such as Microsoft Excel to get a graphical idea of what is going on inside of your robot.

One final advantage of USB serial devices is that they can communicate at higher speeds than traditional serial ports. This means that you can start to transfer image data or other large blocks of data back and forth with your robot with relative ease.

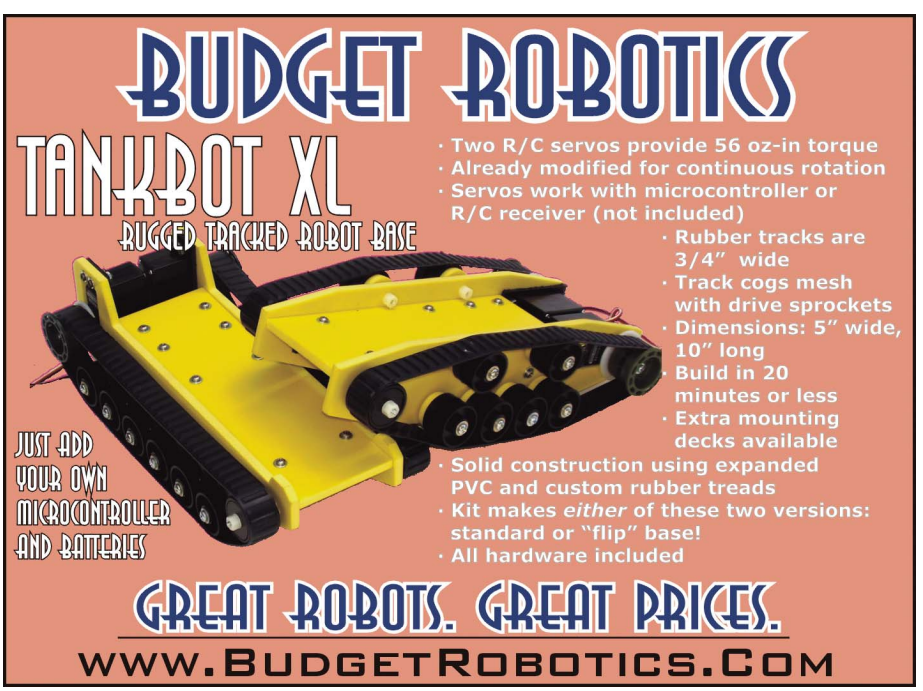
What will you do with these chips? **SV**

RESOURCES

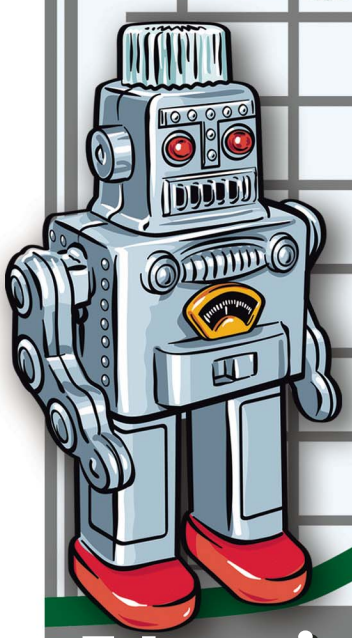
Sparkfun
www.sparkfun.com

Parallax, Inc.
www.parallax.com

Serialterm
www.download.com/SerialTerm/3000-2094_4-10426540.html



Robotic Trends



by Dan Kara

Educational Robotics is the Smart Choice

Following in the footsteps of computers and computer science, robotics will become part of grade school, high school, and university level educational curricula across the industrialized world.

A short press release came across my deck recently that would at first blush seem of little consequence to the robotics community and even less to the world at large. But upon further examination, the piece portends a great deal for roboticists, both of the professional and hobbyist varieties, along with everyone concerned with improving educational systems. For entrepreneurs looking for a sure thing, educational robotics is a godsend.

The press release in question was put out by the Rhode Island Science and Technology Advisory Council (STAC), the Business Innovation Factory, a state run Rhode Island business development group, and FIRST. I am sure many of you have heard of FIRST, but for those who have not, FIRST is a New Hampshire based non-profit organization founded by Dean Kamen (inventor of the Segway) that seeks to inspire an appreciation of science and technology in young people through a variety of robotics competitions. These competitions include the FIRST Robotics Competition and FIRST Vex Challenge for high school students, the FIRST LEGO League for children nine to 14 years old, and the Junior FIRST LEGO League for six- to nine-year-olds.

This unassuming press release announced that Rhode Island will be the first state in the nation to offer a FIRST program, specifically the FIRST Vex Challenge, in all of Rhode Island's 67 public high schools, charter schools, and career and technical centers. Central to the project objectives are plans to integrate the Vex Challenge into the state's new standards for science, technology, mathematics, and engineering (STEM) education.

The Rhode Island program involves the dissemination of VEX robotics kits to each public high school. Students in the program will build robots that will compete in FIRST competitions. FIRST and Rhode Island will monitor the progress of the initiative and make adjustments as necessary. The program is designed to engender an interest in science and engineering using "robotics" as the hook. As such, the initiative is a win-win for FIRST and the State of Rhode Island, whose goals are to increase high school science and engineering proficiency at the national and state (Rhode Island) levels, respectively.

Educational Robotics Proliferates

The Rhode Island/FIRST robotics initiative is interesting as it stands, but when you start parsing through the ramifications of the announcement, it really gets exciting. For example, it is

"Formal inclusion of robotics in educational curricula at all levels worldwide – across multiple disciplines, including 'robotics' itself – is an opportunity that is simply too massive to be missed."

no secret that robotics classes are popping up at school districts across the United States and throughout the industrialized world. Educators in these school districts, like the FIRST and Rhode Island governmental officials, understand the fascination all things robotic have for students, and that this appeal can foster interest in other engineering and scientific disciplines.

What differentiates the Rhode Island program is that they are embracing robotics at the state level. How long will it be before other states take the lead from "little Rhody" and initiate similar programs? Massachusetts, California, and Pennsylvania all consider themselves leaders in the robotics field, and all, like every other state in the US, are looking for ways to increase interest in science and technology among their students. Word travels fast among business development professionals, as well as between those chartered with developing educational curricula at the district, state, and national levels. Taking the Rhode Island/FIRST press release to its logical endpoint, robotics will become central to educational systems across the United States.

It gets even better. Remember, the fascination with robotics transcends national boundaries. Children in Germany, Japan, Korea, India, and Great Britain all are equally captivated by robotics systems. In countries across the world, those responsible for educating the future workforce — a workforce that must compete in an internationalized economy where science and engineering acumen of workers can mean the difference between maintaining standards of living or falling behind — robotics is rightly viewed as a key enabler and educational tool. In these countries, like the US, robotics will become a mainstay of educational curricula at all education levels.

No Matter How You Cut It

It could be argued that the formal incorporation of robotics within school age and university level curricula could take time. That is true. It is equally true, however, that the inclusion will eventually occur. Also, while formal programs are being developed, robotics will continue to be added to school curricula in an unstructured, organic manner. Regardless of how it takes place, the end result will be that robotics will become part of grade school, high school, and university level curricula across the industrialized world.

Personal Computers Provide an Example

The growth and expansive employment of new technology as a teaching tool is not unprecedented. Many of us, and I am dating myself here, can remember a time when personal computers and computer science classes were not found in K-12 schools (or at the university level).

Initially, it was the technology — "computers" — and not the subject — "computer science" — that made inroads into school systems. For those of you old enough to remember, recall that it was individuals that began to bring computers into school environments, often as a novelty. Educators soon found that computer systems made excellent educational facilitators and used them as such at the classroom, departmental, and school level.

For example, the statistical analysis of tree populations in the biology department became much more interesting and fun when computers were used in place of pad and paper techniques. Eventually, computer systems (primarily PC systems) spread from department to department becoming formally incorporated into curricula and employed as educational adjunct for any number of subjects. Employment (and purchase) decisions for classroom computer systems were made at the department, school, state, and even national level.

In my lifetime, "computer science" as a classroom subject has moved from graduate school to preschool. You would be hard pressed to find a school district in the US that does not offer some type of computer class. This changeover differs from simple technological advancement, say, the introduction of PC systems, or when slide rules gave way to calculators, or overhead projectors were replaced by their LCD counterparts. Computer science replaced nothing. It is has become a "field" — like mathematics or biology — and similarly it is part of the educational curricula at the graduate, undergraduate, and K-12 levels throughout the world.

From a historical perspective, computer science then became a separate discipline, but typically part of an engineering or computer sciences department. Finally, computer science became recognized as a core curriculum item worthy of its own department.

Now For Robotics

So much for computers and computer science. Now consider "robotics." Here, we see the same pattern emerging. Consider the following:

• *Organic Growth* — Robots and robotic technology are introduced in the individual classroom or school as a novelty item and teaching tool.

• *Robotics as an Engineering and Science Education Facilitator* — Robots and robotic technology are introduced at the school, district, and state level as

an inducement and facilitator to increase science and engineering proficiency.

• *Robotics as an Engineering and Science Educational Adjunct* — Robots and robotic technology become incorporated into multiple disciplines and departments, including those outside of science and engineering.

• *Robotics as a Separate Discipline* — Robotics is recognized as a separate educational/engineering discipline, but incorporated within other departments (computer science, engineering, etc.).

Dan Kara is President of Robotics Trends, the producer of the RoboBusiness (**www.robobusiness2006.com**) and RoboNexus (**www.robonexus.com**) conferences, and publisher of Robotics Trends (**www.roboticstrends.com**), an online news, information, and analysis portal covering the personal, service, and mobile robotics market. He can be reached at dk@roboticstrends.com

• *Robotics as a Core Curriculum Item* — Robotics becomes a core curriculum item unto itself with its own department.

Already we are witnessing the first two stages of the educational robotics lifecycle, and a handful of universities have reached the other stages. It is clear that we are at the Early Adaptor stage of the educational robotics adoption lifecycle (to borrow a phrase from the Information Technology market). The implication, however, is clear. Robotics — like the computers and computer science before it — will become an integral part of educational systems throughout the world, even to the point of becoming a distinct discipline.

Educational Robotics Everywhere

The use of robotics as a teaching tool and facilitator is the smart choice for educators, along with states and nations, looking to increase science and engineering proficiency of their students (and future workforce). With the Rhode Island/FIRST statement, along with similar announcements made in other industrialized nations, it is clear that the first wave of unstructured, organic incorporation of robotics into school systems is giving way to more formalized approaches. Ultimately, the call for robotics to be included in curricula will be made at the state and even national level.

For entrepreneurs and those looking to robotics as a profession, educational robotics represents a massive market in the earliest stages of growth. The FIRST/Rhode Island press release gives little outright indication of the market potential, but with a little imagination and forethought, the opportunity is made plain. Formal inclusion of robotics in educational curricula at all levels worldwide, across multiple disciplines, including "robotics" itself, is an opportunity that is simply too massive to be missed. **SV**

ROBO-LINKS

Give your Robot the power it deserves...
LORAX WORKS
proSeed microcontroller
www.loraxworks.com

ALL ELECTRONICS CORPORATION
Electronic Parts & Supplies Since 1967

Robotics Kits *and* Components... the World's Most Complete Offering!
WWW.ROBOTSTORE.COM

Technological Arts
9S12 on a DIP!
from $30
freescale Alliance Member
Toll-free: 1-877-963-8996
www.technologicalarts.com

IntelliBrain™-Bot
- Java™ programmable
- Example programs
- Tutorials
- IntelliBrain controller available separately
www.ridgesoft.com

CONTROLLERS
NEW ARM2103 Controller
ONLY $24.90
Includes LPC2103 Microcontroller, Direct In-Circuit Program Download and Update, Easy to connect Port Connections, Delivered Ready to Run, Ideal for Robots and Control Systems.
Also Available: ARM7024 Controller, ATMEGA8535 Controller, and PIC16F877 Controller
www.futurlec.com

Minisumo & Linefollower
Programs in Basic
Kit or RTG
www.snailinstruments.com

$29.95 (MSRP) MaxSonar-EZ1
High Performance Ultrasonic Range Finder
- serial, analog voltage & pulse width outputs
- lowest power - 2mA
- narrow beam
- very easy to use!
www.maxbotix.com

PLANTRACO
CARBON BUTTERFLY
WORLD'S SMALLEST R/C AIRPLANE
READY TO FLY
WWW.PLANTRACO.COM

MADELL TECH
www.madelltech.com
Oscilloscope, Spectrum Analyzer
Function Generator
Rework Station, Reflow Oven
DC Power Supply

DYNOMOTION .COM
MOTION CONTROL W/ AMPS & SOFTWARE
SAVE MONEY NOW

We have your parts!
Kits too!
MAXIMUM ROBOTICS
maximumrobotics.com

Your BASIC Stamp Headquarters
PARALLAX INC
www.parallax.com

RobotShop
Robotics at your service!™
www.robotshop.ca

SERVO MAGAZINE
HIGH TECH AND HANDY
NOW IS THE TIME
To order your subscription to *SERVO Magazine*! Call us TODAY or visit our website to place your order.
877.525.2539 www.servomagazine.com

Amoeba!
www.camppeavy.com

Your One-Stop Robot Shop
The Robot MarketPlace
www.RobotMarketPlace.com

For the finest in robots, parts, and services, go to www.servomagazine.com and click on Robo-Links to hotlink to these great companies.

Scon is Servo Control!
Over 16 Basic-like Scon instructions, 8 High resolution PWM servo outputs, 256 KB Flash memory/32,000 elements, Two Digital outputs & three inputs, and much more!
For further information and to order:
WWW.SCONON.COM

www.arexx.com
AREXX
AREXX Engineering

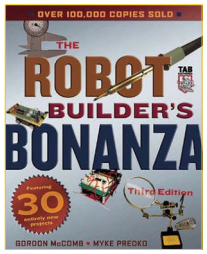
Mind Candy For Today's Roboticist

The SERVO Store

Practical Electronics for Inventors
by Paul Scherz

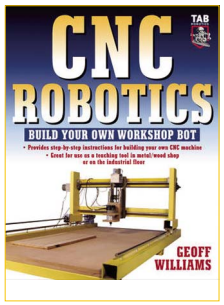
This intuitive, applications-driven guide to electronics for hobbyists, engineers, and students doesn't overload readers with technical detail. Instead, it tells you — and shows you — what basic and advanced electronics parts and components do, and how they work. Chock-full of illustrations, *Practical Electronics for Inventors* offers over 750 hand-drawn images that provide clear, detailed instructions that can help turn theoretical ideas into real-life inventions and gadgets. **$39.95**

Robot Builder's Bonanza Third Edition
by Gordon McComb / Myke Predko



Everybody's favorite amateur robotics book is bolder and better than ever — and now features the field's "grand master" Myke Predko as the new author! Author duo McComb and Predko bring their expertise to this fully-illustrated robotics "bible" to enhance the already incomparable content on how to build — and have a universe of fun — with robots. Projects vary in complexity so everyone from novices to advanced hobbyists will find something of interest. Among the many new editions, this book features 30 completely new projects! **$27.95**

CNC Robotics
by Geoff Williams

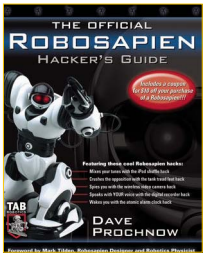


CNC Robotics gives you step-by-step, illustrated directions for designing, constructing, and testing a fully functional CNC robot that saves you 80 percent of the price of an off-the-shelf bot — and that can be customized to suit your purposes exactly, because you designed it. Written by an accomplished workshop bot designer/builder, this book gives you all the information you'll need on CNC robotics! **$34.95**

SERVO CD-Rom

Are you ready for some good news? Along with the first 14 issues of *SERVO Magazine*, all issues from the 2005 calendar year are now available, as well. These CDs include all of Volume 1, issues 11-12, Volume 2, issues 1-12, and Volume 3, issues 1-12, for a total of 26 issues all together. These CD-ROMs are PC and Mac compatible. They require Adobe Acrobat Reader version 6 or above. Adobe Acrobat Reader version 7 is included on the discs. **$24.95 – Buy 2 or more at $19.95 each!**

The Official Robosapien Hacker's Guide
by Dave Prochnow



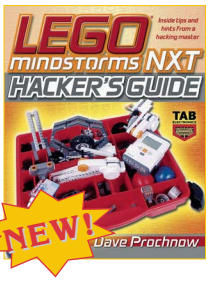
The Robosapien robot was one of the most popular hobbyist gifts of the 2004 holiday season, selling approximately 1.5 million units at major retail outlets. The brief manual accompanying the robot covered only basic movements and maneuvers — the robot's real power and potential remain undiscovered by most owners — until now! This timely book covers all the possible design additions, programming possibilities, and "hacks" not found anyplace else. **$24.95**

PIC Basic Projects
by Dogan Ibrahim

Covering the PIC BASIC and PIC BASIC PRO compilers, *PIC Basic Projects* provides an easy-to-use toolkit for developing applications with PIC BASIC. Numerous simple projects give clear and concrete examples of how PIC BASIC can be used to develop electronics applications, while larger and more advanced projects describe program operation in detail and give useful insights into developing more involved microcontroller applications. **$29.95**

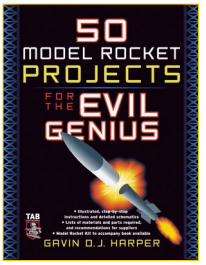
We accept VISA, MC, AMEX, and DISCOVER
Prices do not include shipping and may be subject to change.

LEGO MINDSTORMS NXT Hacker's Guide
by Dave Prochnow



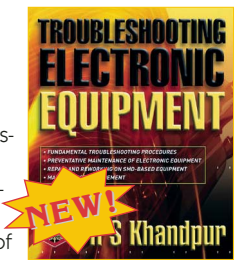
Here is an awesome next-generation collection of LEGO MINDSTORMS projects that enables you to build and program a real working robot in just 30 minutes! New technologies and expanded sensor capabilities make it easier than ever to add a level of sophistication to robotic and architectural creations. This cutting-edge guide describes new advances that make LEGO MINDSTORMS NXT such a great robotics resource. The book explains the all-new NXT intelligent brick ... the interactive servo motors with rotation sensors that align speed for precise control ... the ultrasonic sensor that allows robots to "see" by responding to movement ... the improved light and touch sensors that let robots detect color and feel ... and much more. **$24.95**

50 Model Rocket Projects for the Evil Genius
by Gavin D. J. Harper



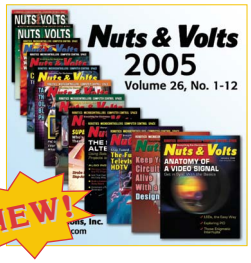
Yes, as a matter of fact, it IS rocket science! And because this book is written for the popular Evil Genius format, it means you can learn about this fascinating and growing hobby while having run creating 50 great projects. You will find a detailed list of materials, sources for parts, schematics, and lots of clear, well-illustrated instructions. **$24.95**

Troubleshooting Electronic Equipment
by R. S. Khandpur



From cell phones to medical instruments to digital and microprocessor based equipment, this hands-on, heavily illustrated guide clearly explains how to troubleshoot, maintain, and repair all types of electrical equipment. The author covers all the essentials such as necessary tools, soldering techniques, testing, fundamental procedures, and mechanical and electrical components. **$49.95**

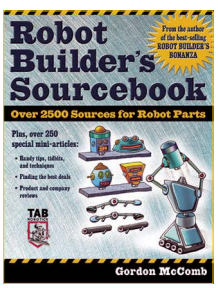
Nuts & Volts CD-Rom



Here's some good news for *Nuts & Volts* readers! Along with all 12 issues of *Nuts & Volts* from the 2004 calendar year, the 2005 issues are now available, as well. These CDs include all of Volumes 25 and 26, issues 1-12, for a total of 24 issues (12 on each CD). These CD-ROMs are PC and Mac compatible. They require Adobe Acrobat Reader version 6 or above. Adobe Acrobat Reader version 7 is included on the discs. **$24.95 – Buy 2 or more at $19.95 each!**

Robot Builder's Sourcebook

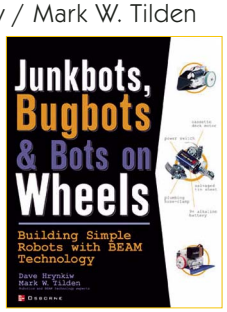
by Gordon McComb



Fascinated by the world of robotics, but don't know how to tap into the incredible amount of information available on the subject? Clueless as to locating specific information on robotics? Want the names, addresses, phone numbers, and websites of companies that can supply the exact part, plan, kit, building material, programming language, operating system, computer system, or publication you've been searching for? Turn to *Robot Builder's Sourcebook* – a unique clearing-house of information for that will open 2,500+ new doors and spark almost as many new ideas. **$24.95**

JunkBots, Bugbots, and Bots on Wheels

by Dave Hrynkiw / Mark W. Tilden

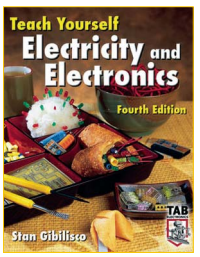


From the publishers of *BattleBots: The Official Guide* comes this do-it-yourself guide to BEAM (Biology, Electronics, Aesthetics, Mechanics) robots. They're cheap, simple, and can be built by beginners in just a few hours, with help from this expert guide complete with full-color photos. Get ready for some dumpster-diving! **$24.99**

Check out our online bookstore at www.servomagazine.com for a complete listing of all the books that are available.

Teach Yourself Electricity and Electronics – Fourth Edition

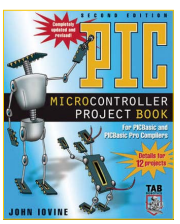
by Stan Gibilisco



Learn the hows and whys behind basic electricity, electronics, and communications without formal training. The best combination self-teaching guide, home reference, and classroom text on electricity and electronics has been updated to deliver the latest advances. Great for preparing for amateur and commercial licensing exams, this guide has been prized by thousands of students and professionals for its uniquely thorough coverage ranging from DC and AC concepts to semiconductors and integrated circuits. **$34.95**

PIC Microcontroller Project Book

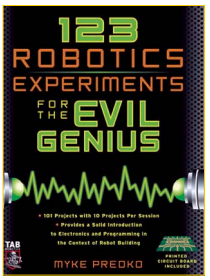
by John Iovine



The PIC microcontroller is enormously popular both in the US and abroad. The first edition of this book was a tremendous success because of that. However, in the four years that have passed since the book was first published, the electronics hobbyist market has become more sophisticated. Many users of the PIC are now comfortable shelling out the $250 for the price of the Professional version of the PIC Basic (the regular version sells for $100). This new edition is fully updated and revised to include detailed directions on using both versions of the microcontroller, with no-nonsense recommendations on which is better served in different situations. **$29.95**

123 Robotics Experiments for the Evil Genius

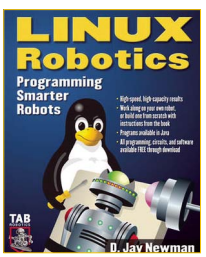
by Myke Predko



If you enjoy tinkering in your workshop and have a fascination for robotics, you'll have hours of fun working through the 123 experiments found in this innovative project book. More than just an enjoyable way to spend time, these exciting experiments also provide a solid grounding in robotics, electronics, and programming. Each experiment builds on the skills acquired in those before it so you develop a hands-on, nuts-and-bolts understanding of robotics – from the ground up. **$25.00**

Linux Robotics

by D. Jay Newman



If you want your robot to have more brains than microcontrollers can deliver – if you want a truly intelligent, high-capability robot – everything you need is right here. *Linux Robotics* gives you step-by-step directions for "Zeppo," a super-smart, single-board-powered robot that can be built by any hobbyist. You also get complete instructions for incorporating Linux single boards into your own unique robotic designs. No programming experience is required. This book includes access to all the downloadable programs you need, plus complete training in doing original programming. **$34.95**

From HomoSapien to RoboSapien

Before R2D2 there was R1D1

SERVO Magazine T-Shirts

Get your very own limited-edition *SERVO Magazine* T-shirt. Shirts come in sizes S, M, L, and XL, and are available in either black or white. All shirts are 100% pre shrunk cotton.

Continued from Page 25

2006 SEATTLE ROBOTICS SOCIETY ROBOTHON

Winning Entry of the 2006 RoboMagellan Contest.

AREXX
AREXX Engineering

WANTED: US DISTRIBUTORS AND DEALERS

For EUROPE's No.1 (Educational) ROBOT KITS

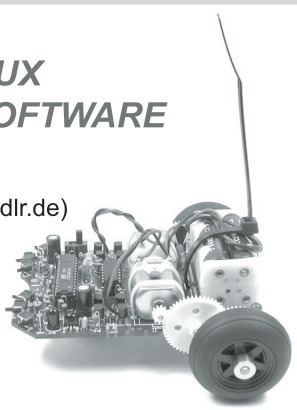
YETI robot kit
Walking robot, programmable in C.
With many extension kits.

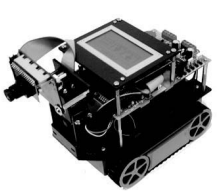
ASURO Educational book.

INCLUDING LINUX SOFTWARE

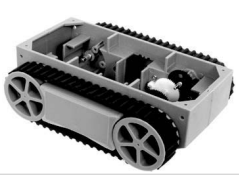
SOCCER robot kit
Ideal for beginners!

ASURO robot kit
Developed by DLR (www.dlr.de)
Programmable in C.





OUR ROBOTS ARE ALSO AVAILABLE AS CHASSIS KITS



ROBBY RP-5 robot kit
Europe's most advanced robot!

WWW.AREXX.COM & WWW.AREXX.NL

More information or interested in distributing these exciting robot bestsellers in the USA?
AREXX Engineering, Nervistraat 16, 8013 RS Zwolle – The Netherlands,
T: +31 38 4542028, F: +31 38 4524482, E: info@arexx.nl, www.arexx.com

An Invitation to Computer Vision With OpenCV

by Robin Hewitt

I attended my first hobbyist robotics club meeting at the Robotics Society of Southern California in 2003. I wasn't sure if I'd enjoy building robots as a hobby, but I knew I wanted to try something new, and the idea of making robots was intriguing. The meeting opened with a presentation about the challenge of implementing visual processing for an entry in the 2004 DARPA Grand Challenge. From that moment, I was hooked. I knew immediately that this was something I wanted to be doing!

Three and a half years later, I'm glad, and grateful, for that serendipitous first exposure to computer-vision and to the challenge of creating vision capabilities for autonomous robots. During the intervening period, enabling robots to see the world around them grew from my weekend hobby into a passion that's re-invigorated my career as a professional programmer and engineer.

It brought me this past summer to Idealab in Pasadena, CA where I was privileged to work with the industry innovators at Evolution Robotics. It's taken me back to college, as well. I'm fortunate to live near UC San Diego, home to several substantial research programs in computer vision, robotics, and computational intelligence. Through an extended studies program, courses at UCSD are open to the general public as well as to regular students. The opportunity to participate in computer-vision research, and to come into contact with the bright young students working on these projects, has been a stimulating experience for this middle-aged, mid-career programmer.

One of the projects I encountered at UCSD was SodaVision. Now here's a wild and crazy idea: Let's give a soda-vending machine the ability to identify people who use it by visual face recognition. Let's also give customers the option of setting up an electronic account that the soda machine can automatically debit. If it "knows" you, it gives you your drink and pays for it out of your account. Too bizarre to be true? Well, guess what ... In fact, see for yourself at **http://sodavision.com**.

Sure, there are more practical projects one might choose to tackle. But hey, this was fun. It also attracted media attention for the students who worked on it. It was featured in Slashdot and on the local news.

If you look at the contributors' page, you'll also see me listed. Those weird-looking views of users' faces on the project's home page were my doing. They're the output of a pre-processing step to improve recognition quality.

In addition to volunteer and student labor, this project might not have been possible without another resource: free, public software. SodaVision was developed with the help of two free software packages: OpenCV for computer vision, and ANN for data retrieval. (URLs for both are listed in the Links box.) The basic video input, face detection, and face recognition capabilities in SodaVision were implemented with functions that are included in OpenCV.

OpenCV is rapidly becoming a standard toolkit for computer-vision research and development at many universities, as well as at UCSD. A number of graduate students use it in their projects, and professors use it in their courses – at both the graduate and undergraduate levels.

No vehicles completed the 2004 Grand Challenge I'd heard about during my intitial introduction to hobby robotics and computer vision. But the following year, several did. The team at Stanford University that built Stanley — the winning entry in the 2005 Grand Challenge — also used OpenCV for computer-vision development.

What about you? Would you like to incorporate computer-vision capabilities into one of your robotics projects? If so, I invite you to join me in 2007 as I take you through a five-part introduction to using OpenCV, starting in January, right here in *SERVO Magazine*. I'll pass on the experience I gained from working on SodaVision, and show you how to use OpenCV to detect and track faces, and even to recognize individuals.

Wishing you joy on all your robotics projects and fun beginnings in 2007! **SV**

LINKS

Robotics Society of Southern California
http://rssc.org

SodaVision
http://sodavision.com

OpenCV
http://sourceforge.net/projects/opencvlibrary

ANN
www.cs.umd.edu/~mount/ANN

Then and NOW

BIONICS — WHERE ROBOTS MEET HUMAN FLESH

by Tom Carroll

"Steve Austin, astronaut. A man barely alive. Gentlemen, we can rebuild him. We have the technology. We have the capability to build the world's first bionic man. Steve Austin will be that man. Better than he was before. Better, stronger, faster."

For those of us who were around in the mid-70s, or who like to watch re-runs of old sci-fi shows, these words still ring in our ears. These words are from the television series *The Six Million Dollar Man*, in which the main character, ex-astronaut Steve Austin is referred to as a bionic man (see Figure 1). This is the source that most people who are not into robot experimentation usually recall hearing the term bionics.

The TV series was based upon the science fiction novel *Cyborg* by Martin Caidin. The opening credits of the show featured actual NASA 1967 footage of a real-life accident of the Northrop M2-F2 lifting body tumbling down a runway. Bruce Peterson piloted the un-powered "lifting body" precursor to the space shuttle and he actually survived reasonably unscathed, although he lost an eye due to an infection acquired while in the hospital.

Austin supposedly was a bit more injured and had both legs, an arm, and an eye replaced by "modern technology." The replaced left eye had a 20.1-1 zoom lens and night vision. The legs allowed Steve to run 60 mph (at least that's what the opening credits scene shows on a speed measuring device). The replaced right arm had the strength of a bulldozer (however strong that is).

Of course, all us technical types ripped apart the mechanics of how Austin was rebuilt, wondering just what type of power source he had within his body to supply this tremendous energy. Dr. Frankenstein's monster of old was strictly biological replacement of human parts so we didn't have any trouble enjoying those early horror movies. Nuclear power was referred to but we all know that modern nuclear power plants just use decaying uranium in the place of coal or oil to boil water in the same old way to turn a turbine. Also, how could he withstand the acceleration of running and jumping? How did he control these amazing devices hooked to his fragile body? What type of internal structure did he have inside his body to transfer the forces from the "bionic" arm to his legs? How did he prevent his powerful right arm from crushing his normal left arm when he did certain feats? We really didn't care one iota; we just eagerly looked forward to one episode after another as our hero, Steve Austin worked for the OSI (Office of Scientific Intelligence) as a secret agent. Hey, it was just fiction, right? We knew that even six billion dollars wasn't enough to create such a man or cyborg. The field of medical bionics was the stuff of doctor's science fiction dreams.

We didn't have the technology back then, but do we actually have it now? We certainly don't have any sort of motors and actuators that fit

Figure 1. Courtesy 1974 ABC TV files.

Figure 2. Monty Reed and his life suit.

Figure 3. Artificial arm with a hook.

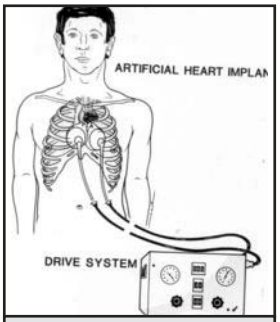


Figure 4. Jarvik-7 artificial heart.

in a human-sized arm yet have the power of a Caterpillar bulldozer. The same goes for a set of human-sized legs that can approach cheetah speeds. Emergency rescue units have "the jaws of life" hydraulic cylinders that can rip apart mangled cars but they have a separate gas engine driving a hydraulic pump for power. These devices cannot fit inside a person's body. In previous issues of *SERVO*, I have discussed how people have become intimate with robots and how we have shared some distinctly human characteristics with our mechanical brethren. They move about our house doing chores for us, can care for us when we are disabled or ill, and can see, hear, and speak, so how much more intimate can they be?

The August '05 issue of *SERVO* featured a friend of mine from the Seattle Robotics Society, Monty Reed and his Lifesuit — a robotic exoskeleton that he fits over his body to allow him, or another person who is disabled, to walk like you or I (Figure 2). Is this bionics? No, not really as it is external to the body and does not replace a missing appendage or organ.

Well, if something like an exoskeleton is not bionics, how about robotics becoming part of us? In medicine, bionics means the replacement or enhancement of organs or other body parts by electro-mechanical systems. It has also been called biomimetics, biognosis, biomimicry, or bionical creativity engineering (that's a mouthful). Some have said it is a short form of biomechanics and others say, "no, it is bio-*electronics*."

Zachary Sterne supposedly coined the term at a conference in Oakland, CA in 1960. Sounds a lot like the controversy about the definition of the word "robot," doesn't it? How do these things differ from typical prostheses? Bionic implants or additions differ from prosthetics by mimicking the original appendage or organ function very closely or even surpassing its capabilities.

Replacement Human Parts

The replacement of parts of the human body with non-living parts is certainly not new. Legs have been lost in battles for millennia and wooden legs have been attached to the body with various types of straps to give the wearer a certain degree of mobility. For centuries, glass eyes, teeth carved from bone and ivory, and arms with hooks have also been attached to the body after loss. What would we have called Captain Hook without his arm prosthesis?

Figure 3 shows a slightly more modern artificial arm with a hook. None of these bodily additions can qualify as bionics as they were just non-moving add-ons to the body. Even the more modern prosthetic arms and legs that have spring-loaded movable joints and re-positionable joints do not qualify. It was the advent of small, efficient, battery-powered electric motors that made true bionic limbs possible.

Artificial Hearts

However, some of the most noteworthy bionic additions to humans were not appendages but replacements for failing or failed human organs. Artificial hearts are the most prominent of these devices, as they are lifesaving implants within the body, not just life-enhancing additions such as arms and legs. Prototype artificial hearts were first developed back in the mid-1950s when Dr. Paul Winchell first patented his artificial heart.

The best known artificial heart is the Jarvik-7 named after its designer, American physician Robert K. Jarvik and was first used in the early 1980s (Figure 4). It functions much like a natural heart and has two pump sections like our own ventricles that pump the blood from the inlet to an outlet valve. Seattle dentist, Barney Clark survived on the Jarvik-7 for 112 days in 1982. He died because of medical complications, not because his artificial heart had failed. A later patient, William Schroeder, lived for 620 days.

This type of artificial heart had two major drawbacks: the need for an external air compressor to drive the heart and the formation of blood clots within the heart. The human heart is a powerful muscle that requires a lot of energy to pump blood throughout our bodies, a big task for a mechanical replacement. We generally think of blood as a red liquid like any other liquid, but it is composed of many types of fragile cells that are easily damaged by mechanical devices. It is this damage that causes stroke-causing clots. Although about 90 people received the Jarvik-7, it was these problems that caused these types of artificial hearts to only be used as temporary life-sustaining devices until an organ

Figure 5. The AbioCor artificial heart.

Figure 6. The Utah Arm.

transplant became available.

Figure 5 shows a second type of artificial heart — the AbioCor — which became available in 2001. Unlike the Jarvik-7, the AbioCor is powered by an internal electric motor and has no connections outside the body, thus reducing the risk of an infection. It has an internal backup battery to drive the heart and that is kept charged by two inductive coils, one just under the skin and the other outside the body to transfer the energy to keep the heart pumping and the internal battery charged.

An external battery pack is worn on a belt or suspenders and enables the patient to be mobile. No artificial heart design has yet been able to use the body's own natural biological energy. The ideal energy source would be an internal biological fuel cell that would convert glucose into power for the artificial heart.

There are several patients who have been kept alive for several years with the AbioCor. A related device — the ventricular assist device (VAD), or artificial ventricle — is an internally implanted pump designed to aid a person with a failing left ventricle. It does not require the removal of the patient's heart as would be with a "typical" artificial heart.

Motor-Driven Appendages

There are many other types of "additions" to the human anatomy that have enhanced people's lives over the years. The cochlear implant within the ear to allow the deaf to hear is another of the more dramatic developments. Artificial retinas, kidneys, pancreas, and heart-lung machines are some of the others, but it is the electrically movable appendage that interests the robotics experimenter the most.

The earliest motorized appendages were arms that had manually controlled, electrically driven hands and wrists. These devices allowed users to grasp objects with the prosthetic hand or claw and to rotate the wrist to allow pouring of a glass of water or similar action. The arm still had to be positioned by the wearer's other natural hand for the best location of the artificial hand. Even some of the newer types of powered prostheses with multiple axes of motion could only operate one joint at a time.

The Utah Arm

One of the first artificial arms controlled by electrical signals from muscles was the Utah Arm and Hand developed in 1981 at the University of Utah by the Center for Engineering Design led by Dr. Steve Jacobsen. In 1987, the Salt Lake City company, Motion Control, released the Utah Arm 2 "with entirely re-engineered electronics that made the Utah Arm one of the most durable and dependable myoelectric arms available," according to the company.

This arm is controlled by muscle (myo-) electrical signals derived from sensors attached to the skin near the stump of an amputated arm. The Utah Arm 3 was released in December of 2002 that had electric elbow, hand, and wrist functions. Five units were placed into field trials on every-day wearers of a transhumeral (higher level) electric arm prosthesis (Figure 6). By early 2004, a total of 27 Beta units of the Utah Arm 3 had been placed and fitted to regular users. The newest version can allow users to simultaneously control both elbow and hand motions.

In early April of this year, a special panel discussion at the Experimental Biology 2006 Conference in San Francisco was titled "The Six Billion Dollar Man," a play on the title of the three decades' old TV series. It focused on the future application of bionics and the rehabilitation of people who had lost limbs in accidents. One of the featured topics was the control of prosthetic arms by thought, not movement of muscles to produce control signals. The Rehabilitation Institute of Chicago's (RIC) Neural Engineering Center for Artificial Limbs (NECAL) has developed such an arm prosthesis. Jesse Sullivan is touted as the world's first "Bionic Man," according to the Rehabilitation Institute of Chicago and his doctors describe him as "the first amputee with a thought-controlled artificial arm" (Figure 7).

In May 2001, Sullivan lost both of his upper extremities as a result of a power line accident. What every lineman fears is not a fall from a tower or pole but coming in contact with tens of thousands of volts at many amps. Both of his arms were burned beyond repair and what was left of them was amputated. After a period of recuperation, RIC began working with Sullivan to apply their amazing thought-controlled arms to his body.

He can now climb a ladder at his house in Dayton, TN, and apply a fresh coat of paint with a paint roller. He can also use a weed-whacker,

Figure 7. Jesse using his bionic arm.

Figure 8. Body bracing for the "bionic" arm.

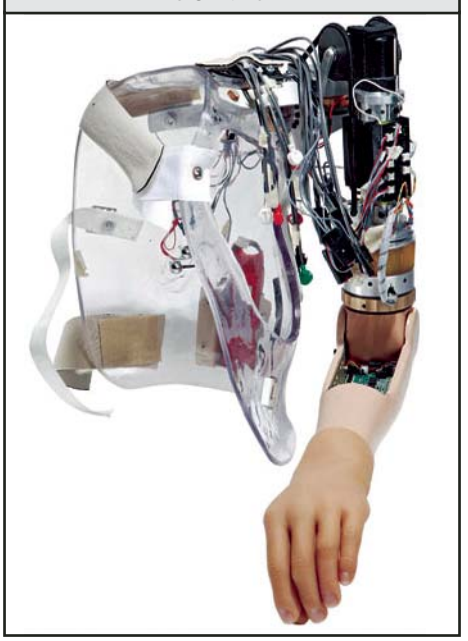


Figure 9. Claudia Mitchell.

bending his elbow and rotating his forearm to cut weeds and grass. And most of all, he is overjoyed to be able to hug his grandchildren. Sullivan can think, "Close my hand," and electrical signals sent through nerves that have been surgically re-routed make it happen. It is a variation of what is called a muscle reinnervation procedure that takes an amputee's own nerves and connects them to a healthy muscle.

Four of Sullivan's nerves were dissected from the shoulder and transferred to the muscles of his chest. Thought-generated nerve impulses are sensed via surface electrodes and carried through to the mechanical arm, causing the arm to move. Today, Sullivan is able to do many of the routine tasks he took for granted before the accident like "putting on socks, shaving, eating dinner, taking out the garbage, carrying groceries, and vacuuming." Yes, the arms are a bit unwieldy and robot-like with motors and wires protruding, but for someone with no ability to manipulate objects, it's "a gift from Heaven." It will be dramatically improved in the future.

The whole assembly weighs just 5.5 kg although body bracing is still required as it cannot yet bolt successfully into the existing bone structure (Figure 8). It does suffer from slow speed and poor durability, but progress is being made and a commercial version is expected to be ready by 2008.

Sullivan is not the only person to make use of this new technology. Figure 9 shows Claudia Mitchell — a woman who lost her arm at the shoulder in a motorcycle accident and has become the fourth person (and first woman) to become the recipient of the thought-controlled arm. An ex-Marine, her accident was not service related but turned her life upside down. She first had to get used to using a standard "hook" type of prosthetic arm, and now she is looking forward to having a new generation arm that will allow her to feel objects. Like Sullivan, she can reach out and grasp a banana just by thinking about the "simple" task. "The first time I peeled a banana one-handed, I cried. I use it to help with cooking, for holding a laundry basket, for folding clothes — all kinds of daily tasks," she says. This new arm is "absolutely amazing" touts the press.

Going from what we had "then" — wooden legs and arms that did not move at all — to life-like arms with mind and microprocessor control that we have "now," well, we've come a long ways. Yes, we have quite a bit of improvement ahead of us and I foresee a few *SERVO* readers involved in this research and development.

As with all my articles, this is just "the tip of the iceberg." I encourage you to dig deeper on the Internet to get to the bottom of each subject that I cover. There are literally millions of links that cover the subject of artificial limbs and bionics. Look back at the November '05 issue of *SERVO* to David Geer's article on "The Shadow Dexterous Hand Knows" for a great piece on an artificial hand. Also, look at a copy of one of the "Evil Genius" series of books entitled *Bionics for the Evil Genius* by Newton Braga. It covers a lot on a bionic ear, vision, and smell sensing, through more sensory than robotic manipulative appendages. It is a fun and inexpensive way to learn bionics via 25 complete projects. There's also some good biology information. **SV**

ABOUT THE AUTHOR

Retired from Rockwell and NASA projects, Tom Carroll is a space robotics engineer. He's authored numerous articles on combat robots, lives on an island in the Pacific Northwest with his wife, Sue, and enjoys robotics, kayaking, and hiking. He can be reached at: TWCarroll@aol.com

Advertiser Index

All Electronics Corp. 48, 81
AREXX Engineering 81, 85
BEST Robotics Competition 71
Budget Robotics 77
COMBOTS 38
CrustCrawler 13
DynoMotion 81
Electronics123 49
Futurlec 81
Hitec 21
Hobby Engineering 9
HobbyLab 49
Homebrewed Robots 81
Jameco Robot Store 2, 81
Lorax Works 48, 81
Lynxmotion, Inc. 3
Maxbotix 81
Maximum Robotics 47, 81
Net Media 91
NU-BOTICS 49
Ortech Education Systems 37
Parallax, Inc. ....Back Cover
Plantraco 49, 81
Pololu Robotics & Electronics 70
RCATS 49
RidgeSoft, LLC 81
RoboteQ 43
Robot Power 39
RobotShop, Inc. 48, 81
SCIENTIFICS 17
Scon Technologies 49, 81
Snail Instruments 81
Solarbotics 21
Solutions Cubed 80
Technological Arts 81
The Robot MarketPlace 32, 81
Vantec 39

Protect the environment
Make the right choice

SerialLCD™

ALAILABLE IN RED, GREEN AND BLUE.

SitePlayer TELNET

PoE - Power Over Ethernet

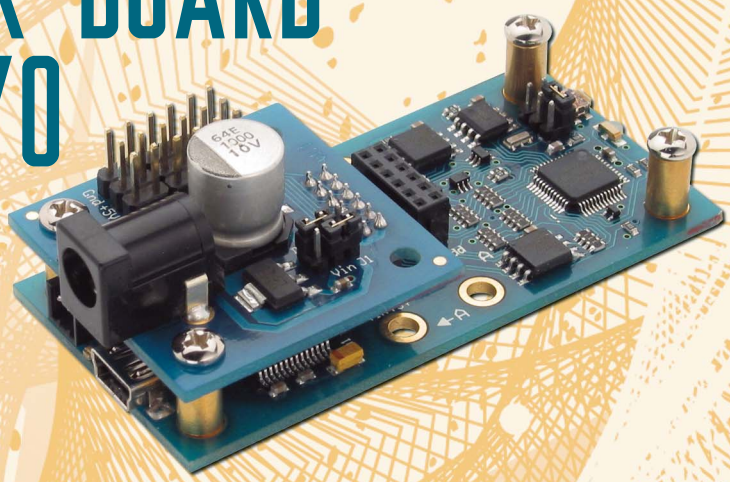
Serial to Ethernet converter that allows your serial device to be controlled via ethernet or through the internet.

$129.95 www.siteplayer.com

RoHS 2002/95/EC **COMPLIANT PRODUCTS FROM NETMEDIA.**

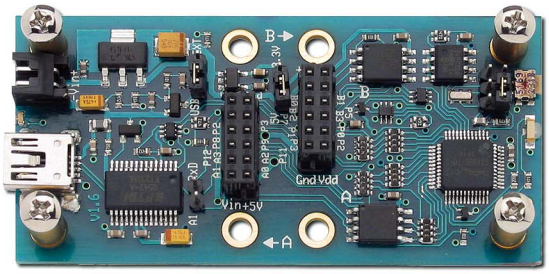
NetMedia Inc., 10940 N Stallard Pl, Tucson Arizona 85737 tel.520.544.4567 fax.520.544.0800 www.BasicX.com

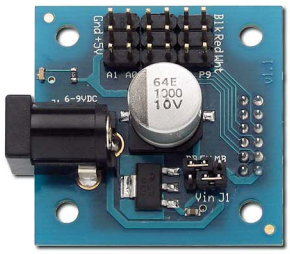
BS2PE MOTHER BOARD AND POWER I/O DAUGHTER BOARD



Parallax's BASIC Stamp® 2pe Motherboard provides a compact, professional-grade platform for BASIC Stamp applications. With the "MoBoStamp-pe" and the coming assortment of plug-in daughterboards, you can integrate and package one-off or multiple application systems with ease. The MoBoStamp-pe has a footprint of less than four square inches. It includes a USB interface for programming and debugging from a PC.

Interface to the outside world takes place through the two daughterboard sockets. Plug-compatible daughtercards can handle interfacing, data acquisition, and sensing functions. The currently-available PWR-I/O-DB card provides an interface for up to six Parallax serial (three-pin) devices and/or servos, and provides separate voltage regulation and filtering for them. Watch for future cards that enable robot navigation and that provide environmental sensing, optical detection and measurement, LED displays, network connectivity, and programmable logic controller (PLC) interfacing, to name just a few. Each motherboard also includes two user-programmable Atmel AVR® microcontrollers. These microcontrollers act as coprocessors to offload computer-intensive and background tasks from the BASIC Stamp, yielding a high performance level while retaining the legendary ease of programming of the BASIC Stamp microcontroller.





The new BASIC Stamp 2pe Mother Board (#28300; $79.95) and **Power I/O Daughter Board** (#28301; $14.95) are sold separately.

Parallax's "PWR-I/O-DB" is a daughterboard that plugs into the BASIC Stamp 2pe Motherboard. It provides both power and a handy interface to potentiometers and servos, along with Parallax's various three-pin sensor devices.

To order online visit www.parallax.com or call the Parallax Sales Department toll-free at 888-512-1024 (Mon-Fri, 7am-5pm, PT).

PARALLAX

www.parallax.com

BASIC Stamp is a registered trademark of Parallax, Inc. AVR is a registered trademark of Atmel.